Rai Studies

Quick Start Site for JAVA Developers

Home | Java | Struts 1 | Spring | MyBatis(iBatis) | Tomcat | About Me

Ajax Form Validation using Spring MVC and JQuery

Posted by Rahul

Go to comments Leave a comment

Like 0 g+1 7 12

In this tutorial, we will see how to validate a form on server side using Ajax with Spring MVC and JQuery. Spring MVC provides a very convenient process for adoption of Ajax with annotation driven configuration. We will use this annotation driven configuration to send Ajax response in form of JSON data. The response will contain state of the form validation and error messages in any error exist in form data.

Tools Used:

- Spring MVC 3.0.3
- JQuery 1.4.2
- Jackson 1.5.3

In the example, we will add users with name and education to a list using Ajax and Spring MVC. The user data will be send to Spring MVC controller with the help of JQuery and the controller will return the full list of users added till time if there is no validation error in form data or the controller will return validation errors if they present in form data.
So, we will learn two important things in this tutorial:

1. **How to validate form data using Ajax in Spring MVC with the help of JQuery?**
2. **and how send back list of objects to Ajax call as response ?**

Domain Class for User
Following is our User domain class that will hold the form data:

```
package com.raistudies.domain;public class User {

    private String name = null;
    private String education = null;

    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getEducation() {
        return education;

    }
    public void setEducation(String education) {
        this.education = education;
    }
 }
```

There are two fields in our User domain object name and education.

Ajax response domain class for sending JSON response

Following is the domain object that will we use to send response:

```
package com.raistudies.domain;

public class JsonResponse {
        private String status = null;
        private Object result = null;
        public String getStatus() {
                return status;
        }
        public void setStatus(String status) {
                this.status = status;
```

Search: type, hit enter

Tutorials

C Jstl Tutorial
Java Tutorial
MyBatis 3 Tutorial
Spring IoC Tutorials
Spring MVC Tutorial
Spring Security Tutorial
Struts Tutorial

Most Popular Tutorials

Ajax Form Validation using Spring MVC and JQuery
Ajax with Spring MVC 3 using Annonations and JQuery
Ajax with Struts Example
DispatchAction in struts example
DynaActionForm Example With Struts
File upload using Spring MVC and annotation configuration
Form Processing in Spring MVC 3 using Annotation
Simple Form Validation with Struts 1 Example

Spring MVC

Hello World Example with annotation driven Spring 3 MVC in Eclipse
Hello World Example with xml configuration Spring MVC in Eclipse
Form Processing in Spring MVC 3 using Annotation
Form Processing in Spring MVC 3 using xml configuration
Ajax with Spring MVC 3 using Annotations and JQuery
Form Validation in Spring MVC 3 using XML Configuration
Form Validation in Spring MVC 3 using Hibernate Validator (JSR 303)
Spring MVC and Json with JQuery
File upload using Spring MVC and annotation configuration
Spring MVC BeanNameUrlHandlerMapping Example

```
        }
        public Object getResult() {
                return result;
        }
        public void setResult(Object result) {
                this.result = result;
        }

}
```

It contain two properties, "status" and "result". "status" field is of type String and will contain "FAIL" or "SUCCESS". "result" will contain the other data that are to be send back to the browser.

UserController.java

```
package com.raistudies.controllers;

import java.util.ArrayList;
import java.util.List;

import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.validation.BindingResult;
import org.springframework.validation.ValidationUtils;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.ResponseBody;

import com.raistudies.domain.JsonResponse;
import com.raistudies.domain.User;

@Controller
public class UserController {
        private List<User> userList = new ArrayList<User>();

        @RequestMapping(value="/AddUser.htm",method=RequestMethod.GET)
        public String showForm(){
                return "AddUser";
        }

        @RequestMapping(value="/AddUser.htm",method=RequestMethod.POST)
        public @ResponseBody JsonResponse addUser(@ModelAttribute(value="user")
                JsonResponse res = new JsonResponse();
                ValidationUtils.rejectIfEmpty(result, "name", "Name can not be
                ValidationUtils.rejectIfEmpty(result, "education", "Educatioan
                if(!result.hasErrors()){
                        userList.add(user);
                        res.setStatus("SUCCESS");
                        res.setResult(userList);
                }else{
                        res.setStatus("FAIL");
                        res.setResult(result.getAllErrors());
                }

                return res;
        }

}
```

showForm() is used to shoe the Add user form to browser. method **addUser()** is will handle the ajax request and validate the User object and if there is no validation error in form data it will set the **userList** object to result property of JsonResponse class with status as "**SUCCESS**". But, if there are errors in form data it will extract list of errors from **BindingResult** object using **getAllErrors()** method and set to result property of JsonResponse class with status as "**FAIL**".

AddUser.jsp

Following is the jsp page that calls the UserController using Ajax with the help of JQuery:

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.o
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Add Users using ajax</title>
<script src="<%=request.getContextPath() %>/js/jquery.js"></script>
<script type="text/javascript">
        var contexPath = "<%=request.getContextPath() %>";
</script>
<script src="<%=request.getContextPath() %>/js/user.js"></script>
<link rel="stylesheet" type="text/css" href="<%=request.getContextPath() %>/sty
</head>
<body>
<h1>Add Users using Ajax ........</h1>
        <table>
                <tr><td colspan="2"><div id="error" class="error"></div></td></
```

Recent Tutorials

- JQuery Selectors – by id, name or tag
- Spring MVC 3 CRUD example with MyBatis 3
- Spring MVC 3 controller for performing CRUD operation using MyBatis(iBatis) 3
- Ajax Form Validation using Spring MVC and JQuery
- Integrating MyBatis 3 and Spring frameworks – Part 2

```
                <tr><td>Enter your name : </td><td> <input type="text" id="name
                <tr><td>Education : </td><td> <input type="text" id="education"
                <tr><td colspan="2"><input type="button" value="Add Users" oncl
                <tr><td colspan="2"><div id="info" class="success"></div></td><
        </table>
</body>
</html>
```

The jsp page includes a js file **user.js** that has been used for holding the definition of the JavaScript method **doAjaxPost()** which produces the ajax class and also use the response of the Ajax call to dynamically update the page data.

user.js

Following is the code for ajax class and interpreting the response received from Spring MVC controller:

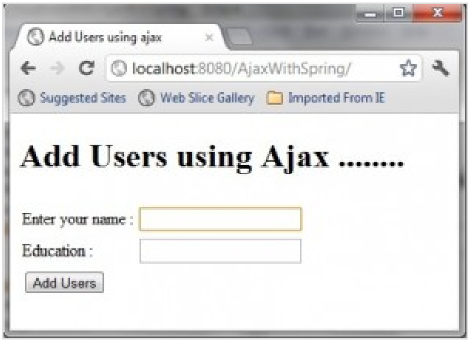
```
function doAjaxPost() {
    // get the form values
    var name = $('#name').val();
    var education = $('#education').val();

    $.ajax({
        type: "POST",
        url: contexPath + "/AddUser.htm",
        data: "name=" + name + "&education=" + education,
        success: function(response){
            // we have the response
            if(response.status == "SUCCESS"){
                userInfo = "<ol>";
                for(i =0 ; i < response.result.length ; i++){
                    userInfo += "<br><li><b>Name</b> : " + response.result[i].n
                    ";<b> Education</b> : " + response.result[i].education;
                 }
                 userInfo += "</ol>";
                 $('#info').html("User has been added to the list successfully.
                 $('#name').val('');
                 $('#education').val('');
                 $('#error').hide('slow');
                 $('#info').show('slow');
             }else{
                 errorInfo = "";
                 for(i =0 ; i < response.result.length ; i++){
                     errorInfo += "<br>" + (i + 1) +". " + response.result[i].c
                 }
                 $('#error').html("Please correct following errors: " + errorIn
                 $('#info').hide('slow');
                 $('#error').show('slow');
             }
         },
         error: function(e){
             alert('Error: ' + e);
         }
    });
}
```

$.ajax() method of JQuery has been used for making an Ajax call here which sends the form data name and education field values to Spring Controller. On success of the Ajax call, it first checks the status field of the response. Note that, the response object here is the JSON representation of the JsonResponse java object.

If the status field of the response is “SUCCESS” then it iterates through the user list using the notation response.result[i] , note that the list object of java is converted to json array by Jackson library.

If the status is “FAIL” the the result object will contain the validation errors which we can access using the notation **response.result[i].code** , here the code will return error message added in Spring controller.

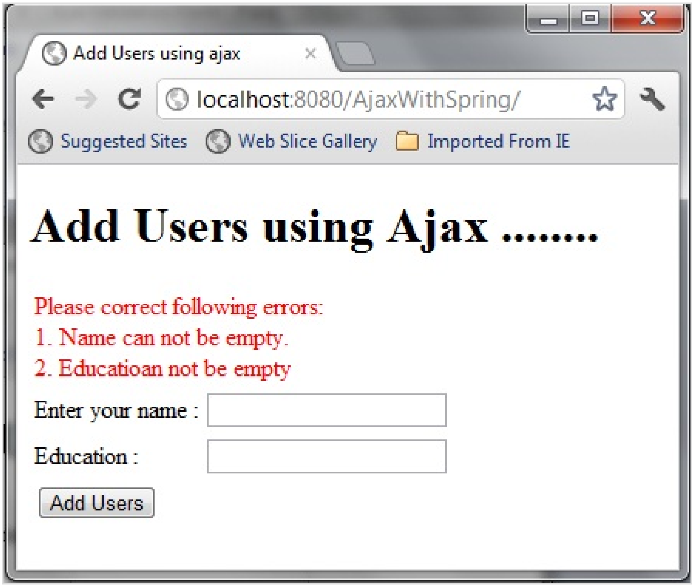
While running the example in tomcat 6 server, it will open following form :



Add Users using Ajax ....

Ajax Validation Form

Just click on the “Add Users” button without entering any value, it will show errors for the field as shown in the bellow image:

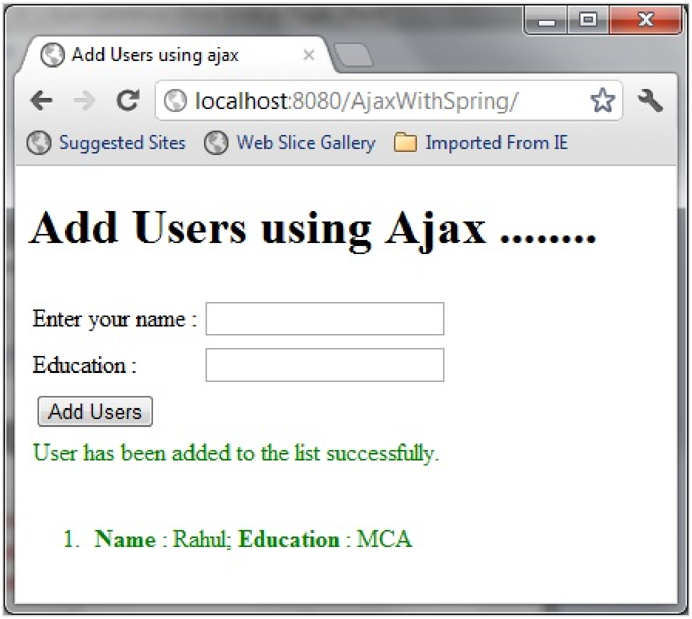


Add Users using Ajax ....

Please correct following errors:
1. Name can not be empty.
2. Educatioan not be empty

Ajax Validation Show Error in Page

Now enter any name and education and click on the “Add Users” button. It will add the user detail to list and also show the info of the whole user list on the page :



Add Users using Ajax ....

User has been added to the list successfully.

1. **Name** : Rahul; **Education** : MCA

Ajax Validation with Spring MVC and JQuery Success Page

You can also try the example by downloading the example from bellow links:

Source: Download

Source + Lib :Download

- Post to Delicious
- Post to Facebook

12

Like 0 +1 7 12

Related Posts:

- Spring MVC and Json with JQuery
- Form Validation in Spring MVC 3 using Hibernate Validator (JSR 303)
- Form Validation in Spring MVC 3 using XML Configuration
- Ajax with Spring MVC 3 using Annotations and JQuery
- Spring MVC 3 CRUD example with MyBatis 3

Ajax, Form Validation, MVC, Spring

Leave a comment ? 14 Comments.

omarps February 27, 2012 at 12:03 am Reply

Hi,

Why aren't you using client-side validation instead of the made of ajax request to form validation.
Even if you choose to validate on both sides, why aren't you using a Validator object?

Regards,
Omar

Rahul Mondal February 27, 2012 at 8:30 am Reply

The motive of this tutorial is to use Spring validation through Ajax call.
And for the second one, yes you are right, we should use implementation of Validator Interface to validate a form object.

Kevinsheng April 14, 2012 at 4:28 am Reply

I don't see where JACKSON is used in the example, however when copying ther libs to lib folder, everything starts working. what class in jackson is being used in the example?

thanks/kevin

Adofo May 22, 2012 at 10:29 am Reply

Hi Kevin,

I have the same question, did you get the answer as to how Jackson is being used.
I will be kind of you to share it 🙂

Kind Regards
Adofo

Rahul June 3, 2012 at 9:00 am Reply

Jackson is being used internally by spring mvc to convert java object to json format.
You just have to put jackson jars to use json in spring mvc.

akasaramanna April 18, 2012 at 3:32 pm Reply

captcha code is static

kkuldeep April 24, 2012 at 11:50 am Reply

This tutorial rocks for me after following adaptations (I used Spring 3.1.1):
1. For Jackson (mainly used for mapping) you need to download the following jars:
jackson-core-asl-1.9.6.jar, jackson-mapper-asl-1.9.6.jar
2. In web.xml, change to *.do here i.e. *.do

3. In UserController.java, index.jsp do change 'AddUser.htm' to 'AddUser.do'

Meryem May 7, 2012 at 4:38 pm Reply

Hi,

It is possible to use validation through Ajax call for a file input ? and how to ?
PS: I want to upload only images in a file input.

Rahul May 16, 2012 at 9:57 am Reply

We can send the file type of the selected file using JavaScript and then check the type on server side.

Jyothi May 17, 2012 at 3:35 pm Reply

hi,

here I am not getting message in simple html format. Message is repeating and displaying in the Json format like below.

{"status":"FAIL","result":[{"codes":["Project Name can not be
empty..projectAttribute.name","Project Name can not be empty..name","Project Name can
not be empty..java.lang.String","Project Name can not be
empty."],"arguments":null,"defaultMessage":null,"objectName":"projectAttribute","field":"name","rejected
Name can not be empty."},{"codes":["Description can not be
empty..projectAttribute.description","Description can not be
empty..description","Description can not be empty..java.lang.String","Description can not
be
empty."],"arguments":null,"defaultMessage":null,"objectName":"projectAttribute","field":"description","rej
can not be empty."}]}

What could be the problem?

Jyothi May 18, 2012 at 3:08 pm Reply

Please help me . I am not able to resolve this issue 🙁

Yogesh July 5, 2012 at 6:48 pm Reply

thank u so much sir for helping me using ajax with spring

online mistress August 15, 2012 at 1:15 pm Reply

Hmm is anyone else experiencing problems with the pictures on this blog loading? I'm trying to figure out if its a problem on my end or if it's the blog.

Any suggestions would be greatly appreciated.

Tanu December 12, 2012 at 7:26 am Reply

Nice tutorial, thanks.

Leave a Comment

NAME

EMAIL

Website URL

NOTE - You can use these HTML tags and attributes:
`<a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code> <del datetime=""> <em> <i> <q cite=""> <strike> <strong>`

Type the text

Privacy & Terms

SUBMIT

Tutorials

- C Jstl Tutorial
- Java Tutorial
- MyBatis 3 Tutorial
- Spring IoC Tutorials
- Spring MVC Tutorial
- Spring Security Tutorial
- Struts Tutorial

Important Links

- MyBatis Official Website
- Spring IoC Documentation
- Spring MVC 2.5.6 Documentation
- Spring MVC 3 Documentation
- Spring Security Documentation
- SpringSource Official Site
- Struts 1 official site

Important Tags

Ajax CRUD Database Eclipse File Upload Form Validation iBatis Insert Integration IoC JQuery Json jstl MVC MyBatis Spring Spring Security Struts 1 Theme Tomcat

Together We Can Win The World

Hi Friends, I am just trying to help you and also help myself. I like to learn new things and make others learn also. If you get any error in my site, please inform me. Thanks Rahul Mondal

^ Top