

ajax Validation trong Struts 2

Struts 2 cung cấp sự hỗ trợ cho AJAX Validation. Trong trường hợp này, page sẽ không refresh hoặc reload, vì thế điều này làm tăng hiệu suất. Nó được thực hiện ngầm định bởi sử dụng JavaScript, ví dụ: được sử dụng cho trình Validation ở Client-Side.

Để sử dụng AJAX Validation, bạn cần thêm **dojo** plugin trong project của bạn.

Giới thiệu jsonValidation Interceptor

AJAX Validation được thực hiện bởi jsonValidation Interceptor. Nó không được tìm thấy trong default stack, vì thế bạn cần định nghĩa nó một cách tường minh. Chính nó không thực hiện bất cứ trình validation nào, và đó là tại sao nó phải được sử dụng với Validation Interceptor. Nó được tìm thấy trong jsonValidationWorkflowStack, mà bao gồm jsonValidation, validation và workflow interceptor và basicstack.

Các bước để thực hiện AJAX Validation

- Tạo form để lấy input từ người dùng.
- Kế thừa lớp ActionSupport trong action của bạn.
- Định nghĩa validation trong validation.xml file.
- Định nghĩa result name cho thông điệp lỗi và đăng ký jsonValidationWorkflowStack trong struts.xml file.

Ví dụ để thực hiện AJAX Validation

Trong ví dụ này, chúng ta tạo 4 page, như sau:

Tạo index.jsp

Cho input từ người dùng. Nó nhận name, password, và email id từ người dùng.

```
<%@ taglib uri="/struts-tags" prefix="s"%>
<%@ taglib uri="/struts-dojo-tags" prefix="d"%>
<html>
<head>
<d:head/>
</head>
```

```
<body>
<marquee>Registration Form.....</marquee>

<s:form action="register">
<s:textfield name="name" label="Username"></s:textfield>
<s:textfield name="email" label="Email ID"></s:textfield>
<s:password name="password" label="Password"></s:password>
<d:submit validate="true" type="image" src="register-now.jpg">
</d:submit>
</s:form>

</body>
</html>
```

Tạo lớp action

:

Lớp này kế thừa lớp ActionSupport và ghi đè phương thức validate.

RegisterAction.java

```
package mypack;

import com.opensymphony.xwork2.ActionSupport;

public class Register extends ActionSupport{
private String name,password,email;

//Phuong thuc setter va getter

public String execute(){
    return "success";
}
}
```

Tạo validation file

Tại đây chúng ta đang sử dụng bundled validator để thực hiện trình validation.

Register-validation.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE validators PUBLIC
  "-//OpenSymphony Group//XWork Validator 1.0.2//EN"
  "http://www.opensymphony.com/xwork/xwork-validator-1.0.2.dtd">

  <validators>

    <field name="name">
      <field-validator type="requiredstring">
        <message>Name không được để trống</message>
      </field-validator>
    </field>

    <field name="email">
      <field-validator type="requiredstring">
        <message>Email ID không được để trống</message>
      </field-validator>
      <field-validator type="email">
        <message>Xin nhập một email id hợp lệ</message>
      </field-validator>
    </field>

    <field name="password">
      <field-validator type="requiredstring">
        <message>Password không được để trống</message>
      </field-validator>
      <field-validator type="stringlength">
        <param name="minLength">5</param>
```

```
<param name="maxLength">10</param>
<message>Password không thể ít hơn 5 hoặc dài hơn 10</message>
</field-validator>

</field>

</validators>
```

Tạo struts.xml

xml file định nghĩa một result bởi tên đã nhập, và một interceptor là jsonValidatorWorkflowStack.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts
Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">
<struts>

<package name="a" extends="struts-default">

<action name="register" class="mypack.Register">
<interceptor-ref name="jsonValidationWorkflowStack"></interceptor-ref>

<result name="success">/welcome.jsp</result>
<result name="input">/index.jsp</result>
</action>

</package>
</struts>
```

Tạo các thành phần view

JSP file đơn giản này hiển thị thông tin về người dùng.

welcome.jsp

```
<%@ taglib uri="/struts-tags" prefix="s" %>
```

```
Welcome, <s:property value="name"/>
```