



TinyMCE using Java for spell check

TinyMCE using Java for spell check

📅 Thursday, February 11, 2010

Recently I was working on a web base project which uses java/j2ee technology. In the project we required a rich text editor with lots of functionality. So we decided to use TinyMCE.

TinyMCE is a platform independent web based Javascript HTML WYSIWYG editor control released as Open Source under [LGPL](#) by [Moxiecode Systems AB](#). It has the ability to convert HTML TEXTAREA fields or other HTML elements to editor instances. TinyMCE is very easy to integrate into other Content Management Systems.

TinyMCE comes with a lots of plugins including a spellchecker.

This post is about the situation where you want to use your own implementation. In that case we can configure "spellcheck" plugin to use our own service.



About Me



[Shrinivas](#)

[View my complete profile](#)



Spell Checker for tinyMCE using Java

Spell Checker using Java and TinyMCE



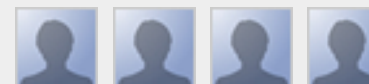
Followers



Join this site

with Google Friend Connect

Members (4)



Below steps shows how to implement spellcheck plugin working with your java base application.

Step 1:

Download tinyMCE version 2.1 from here [TinyMCE](#) and extract it.

Step 2:

Download jazzy-0.5.2.bin.zip and english.0.zip from [Jazzy](#).
Extract both the zip files.

Step 3:

Create a new Web project and name it as Testspellcheck

Step 4:

Now add a folder "jsp" to your project root.

Step 5:

Add a folder "js" to your project root.

Step 6:

Copy /tinymce/jscripts/tiny_mce folder to "js" folder which you created in Step 4.

Step 7:

Copy the jazzy-core.jar file (which can be found in extracted jazzy-0.5.2.bin zip mentioned in Step 2) to /WEB-INF/lib/ in your project.

Step 8:

Already a member? [Sign in](#)



Search This Blog

Search



Your Title



Blog Archive

▼ 2010 (1)

▼ February (1)

[Recently I was working on a web base project whic...](#)

Create a folder "dictionary" under /WEB-INF.

Step 9:

Copy "english.0" file (which can be found in extracted english.0.zip mentioned in Step 2) to newly created folder as mentioned in Step 10.

Step 10: Now add a new "check.jsp" in the /jsp folder.

```
<%
response.setContentType("text/xml");
%>
<%@page import="spellchecker.*"%>
<%
String cmd = request.getParameter("method");
String id = "content"; //request.getParameter("id");
String check = request.getParameter("check");
String lang = request.getParameter("lang");
MySpellChecker sc = MySpellChecker.getInstance(lang);
String suggestionsStart="{\"result\":[";
String suggestionsBody = "";
String suggestionsEnd="]}";
String returnSuggestions = "";
if ("checkWords".equalsIgnoreCase(cmd))
{

```

Step 10:

Under your source folder create package "spellchecker".

Step 11:

--- --

Add two classes "MySpellChecker.java" & "StringListener.java" under the package "spellchecker".

MySpellChecker.java

```
package spellchecker;

import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.*;

import com.swabunga.spell.engine.Configuration;
import com.swabunga.spell.engine.SpellDictionaryHashMap;
import com.swabunga.spell.event.*;

public class MySpellChecker
{
```

StringListener.java

```
package spellchecker;

import java.util.Vector;
```

```

import com.swabunga.spell.event.SpellCheckEvent;
import com.swabunga.spell.event.SpellCheckListener;

public class StringListener implements SpellCheckListener

{
    private Vector errorWords = new Vector();

    public void spellingError(SpellCheckEvent event)
    {
        String word = event.getInvalidWord();
        if (word.length() > 1 && !errorWords.contains(word)
        , ... ..

```

Step 12:

Now create a "test.jsp" under /jsp folder to test your TinyMCE editor.

```

<script type="text/javascript" src="js/tiny_mce/tiny_mce.js"> </script>
<script type="text/javascript" src="js/tiny_mce/plugins/spellchecker/editor_plugin.js"> </script>
<script type="text/javascript">
tinyMCE.init({
    mode : "textareas",
    theme : "advanced",
    plugins : "spellchecker",
    theme_advanced_buttons3_add : "spellchecker",
    spellchecker_languages : "+English=en"

```

```
});  
</script>
```

```
<form method="post" action="somepage">
```

Step 13:

Now open the editor_plugin.js file from
"js/tiny_mce/plugins/spellchecker/" & make the following changes.

Find: `_sendRPC:function (m, p, cb)`

Under that replace

```
editor.getParam("spellchecker_rpc_url","{backend}");
```

with

```
editor.getParam("spellchecker_rpc_url",  "../jsp/check.jsp  
?method="+m+"&check="+p[1]+"<="+p[0]);
```

Note : m, p, cb are the parameters, if naming convention
is
different then make the changes accordingly.

Step 14:

Save the file and restart your web server and try it.

 Posted by Shrinivas at [5:53 AM](#) [0 comments](#)

Subscribe to: [Posts \(Atom\)](#)