ActionFlow Plugin

Name	ActionFlow Plugin
Publisher	
License	Open Source (ASL2)
Version	2.3.4
Compatibility	Struts 2.3.4 +
Homepage	https://github.com/aleksandr-m/struts2-actionflow

{rate:title=Rating|theme=dynamic}

Overview

A Struts2INLINE

plugin for creating wizards (action flows)

.

Features Overview

- Simple integration to new or existing Struts2 application
- Automatic use of Post/Redirect/Get pattern to avoid duplicate form submissions
- Proper handling of browser back and refresh buttons
- Action flow scope to keep data, there is no need to use scoped model-driven actions

Showcase

Showcase application could be downloaded from the Maven Central Repository.

Download struts2-actionflow-showcase

Contributing

Found a bug or have a feature request? Create a new issue or submit a Pull Request.

Questions

If you have questions about how to use `struts2-actionflow-plugin` create a new issue or ask a question on Stack Overflow.

Installation

Copy struts2-actionflow-plugin-x.x.x.jar into your classpath `WEB-INF/lib`. No other files need to be copied or created.

If you are using Maven, add this to your project POM:

xml<dependencies> ... <dependency> <groupId>com.amashchenko.struts2.actionflow</groupId> <artifactId>struts2-actionflow-plugin< /artifactId> <version>2.4.0</version> </dependency> ... </dependencies>

Example Usage

1. Install it by adding this plug-in dependency to your POM or by copying jar into /WEB-INF/lib directory.
2. Make your action package extend `actionflow-default` package.
3. Add `<param name="actionFlowStep">` parameters to actions you want to include in action flows. (NOTE: the action must have an input result!)
4. Use `next` and `prev` actions in JSP to move between wizard steps.
5. Use `@ActionFlowScope` annotation on action classes and fields in order to keep data in action flow scope.

Action Mappings

xml<package name="actionflow-showcase" namespace="/" extends="actionflow-default"> <action name="saveName" method="saveName" class="com.example.FlowAction"> <param name="actionFlowStep">1</param> <result name="input">/WEB-INF/pages/name.jsp</result> <result name="error">/WEB-INF/pages/name.jsp</result> <result>/WEB-INF/pages/name-success.jsp</result> </action> <action name="savePhone" method="savePhone" class="com.example.FlowAction"> <param name="actionFlowStep">2</param> <result name="input">/WEB-INF/pages/phone.jsp</result> <result name="error">/WEB-INF/pages/phone.jsp</result> <result>/WEB-INF/pages/phone-success.jsp</result> </action> </package>

Form

xml<s:form action="next"> <s:hidden name="step" value="%{#session['actionFlowPreviousAction']}" /> <s:textfield key="name" label="Name" /> <s:submit value="previous" action="prev" /> <s:submit value="next" action="next" /> </s:form>

Note: Since Struts2 version 2.3.15.3 if you are using `<s:submit>` tags with `action` attribute you need to enable support for `action:` prefix.

Put that in your struts.xml file:

xml<constant name="struts.mapper.action.prefix.enabled" value="true" />

Action

java@ActionFlowScope public class FlowAction extends ActionSupport { @ActionFlowScope private String name; }

Showing action flow steps in JSP

Available from struts2-actionflow-plugin 2.1.0

Implement `ActionFlowStepsAware` interface in action and create getter for `ActionFlowStepsData`:

javapublic class FlowAction extends ActionSupport implements ActionFlowStepsAware { private ActionFlowStepsData stepsData; @Override public void setActionFlowSteps(ActionFlowStepsData stepsData) { this.stepsData = stepsData; } public ActionFlowStepsData getStepsData() { return stepsData; } }

In JSP iterate over `ActionFlowStepsData#steps` map. Use `#key` and `#value` to get step index (starting from 1) and action name. The `ActionFlowStepsData#stepIndex` property holds index of current step.

xml <s:iterator value="stepsData.steps"> <s:if test="stepsData.stepIndex > key"> <s:set var="status" value="'passed'"/> </s:if> <s:elseif test="stepsData.stepIndex == key"> <s:set var="status" value="'active'"/> </s:elseif> <s:else> <s:set var="status" value="'simple'"/> </s:else> <li class="<s:property value="#status"/>"> <s:property value="key"/> <s:property value="value"/> </s:iterator>

Controlling action flow

Available from struts2-actionflow-plugin 2.3.0

Implementing *ActionFlowAware* interface gives you ability to change flow of actions.

The *nextActionFlowAction* method controls which flow action will be executed next. Return the name of the flow action which should be executed after the action

which is passed as *currentActionName* argument. E.g. if wizard consists of three actions: 'saveName' > 'savePhone' > 'saveEmail', and

action 'savePhone' must be skipped return 'saveEmail' from this method when method argument *currentActionName* is 'saveName'.

On returning not a flow action name or *null*, action flow won't be changed (i.e. the configured next action from the flow will be executed).

The action properties values passed for the current action will be available in *nextActionFlowAction* method.

As a result of that you can skip actions based on user input and current action name.

javapublic class FlowAction extends ActionSupport implements ActionFlowAware { private String name; @Override public String nextActionFlowAction(String currentActionName) { String action = null; if ("saveName".equals(currentActionName) && "skip".equals(name)) { action = "saveEmail"; } return action; } }