

Escenic Content Engine
Struts Form Reference

6.7.8-1

Table of Contents

1 Introduction	3
2 Action Forms	4
2.1 AbstractForm	4
2.2 AbstractUserForm	4
2.3 Profiles	5
2.4 Profile	6
2.5 DefaultUserForm	6
2.6 StandardUserForm	7
2.7 LoginForm	7
2.8 PasswordForm	8
2.9 ValidateUserPasswordForm	9
2.10 ChangePasswordForm	9
2.11 DeleteProfileForm	10
2.12 SearchForm	11
2.13 ArticleSearchForm	12
2.14 SimpleSearchForm	13
2.15 AdvancedSearchForm	14
3 Actions	17
3.1 AbstractProfileAction	17
3.2 AddUserAction	17
3.3 LoginAction	18
3.4 NewPasswordAction	19
3.5 SetPasswordAction	20
3.6 AuthenticatedAction	20
3.7 AuthenticatedForwardAction	21
3.8 PopulateUserAction	22
3.9 UpdateUserAction	22
3.10 DeleteUserAction	23
3.11 DeleteProfileAction	24
3.12 LogoffAction	25
3.13 SearchAction	25

1 Introduction

This reference should contain enough information to enable a template developer to set up profile- and search-related Struts actions in an Escenic publication, so that profile/search forms may be submitted. A working knowledge of Struts is assumed.

For more on Struts, see <http://struts.apache.org>.

2 Action Forms

2.1 AbstractForm

A base class which has behaviour and properties which are common to all action forms. All action forms share the properties defined by this class.

Properties

AbstractForm inherits properties from:

- `com.escenic.tools.struts.ValidateForm`

It has no properties of its own.

2.2 AbstractUserForm

A form that collects user data needed when registering or updating a profile

The properties describes below are all settable from the html form in the jsp page.

The form contains information about the profile to create or update

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.AbstractUserForm"
  type="com.escenic.profile.presentation.struts.AbstractUserForm"/>
```

Action Error

This form requires that `userName` and `surName` is set. If one is missing, the user should be notified.

```
no_userName
  (On the userName property.) The userName was not specified
invalid_username
  (On the userName property.) The username did not start with a letter
no_surName
  (On the surName property.) The surName was not specified
```

AbstractUserForm inherits properties from:

- `com.escenic.profile.presentation.struts.AbstractForm`

It also has the following properties of its own:

userName

String

the username of the user

firstName

String

the firstame of the user

surName

String

the users surname

profiles

Profiles

the profiles, containing additional user information

address

String

the users address

phone

String

the users phone number

mobilePhone

String

the users mobile phone

email

String

the users email

publicationId

String

the publicationid of the profile. Returns -1 if it's a new profile

password

String

the password

id

String

the id of the profile. Returns -1 if it's a new profile

2.3 Profiles

A helper class to facilitate handling of several Profile objects. A Form may use a Profiles instance to handle lists of profiles.

Properties

Profiles inherits properties from:

- `java.lang.Object`

It also has the following properties of its own:

item

Profile

Returns the Profile instance given the name. It is meant to be used from a JSP page, when using Map properties: For example `form.Profiles(1).name` would refer to the name of the profile. If the item does not exist, it will be created. This method, therefore, never returns null.

keys

List

returns a list of keys that can be used to retrieve profiles. The List contains the names that can be used to retrieve each profile.

size

int

returns the total count of Profile objects

2.4 Profile

A profile related to the user

Properties

Profile inherits properties from:

- `java.lang.Object`

It also has the following properties of its own:

id

String

the id of the profile. Is -1 if the profile is not yet created

attribute

String

the attribute defined by the given key

attributeNames

Set

a set containing all the names of attributes added to the profile

articleType

String

the articletype of the profile

homeSectionId

String

the id of the homesection of the profile

2.5 DefaultUserForm

A default implementation of the `AbstractUserForm`

Defines that the email should be used as the username of the user

The form contains information about the user to create or update

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.DefaultUserForm"
  type="com.escenic.profile.presentation.struts.DefaultUserForm"/>
```

DefaultUserForm inherits properties from:

- `com.escenic.profile.presentation.struts.AbstractUserForm`

It has no properties of its own.

2.6 StandardUserForm

A standard implementation of the `AbstractUserForm`

Provide the same functionality as `DefaultUserForm`, except that this form provide a separate user name that could be anything, not just the mail-address.

The form contains information about the user to create or update

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.StandardUserForm"
  type="com.escenic.profile.presentation.struts.StandardUserForm"/>
```

StandardUserForm inherits properties from:

- `com.escenic.profile.presentation.struts.AbstractUserForm`

It has no properties of its own.

2.7 LoginForm

A form that collects user data needed when logging in a user

The properties describes below are all settable from the html form in the jsp page.

The form contains information about the `userName` and password of the user.

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.LoginForm"
  type="com.escenic.profile.presentation.struts.LoginForm"/>
```

Action Error

This form requires that `userName` and `password` is set. If one is missing, the user should be notified.

no_userName
(On the `userName` property.) The `userName` was not specified

no_password
(On the `password` property.) The password was not specified

`LoginForm` inherits properties from:

- `com.escenic.profile.presentation.struts.AbstractForm`

It also has the following properties of its own:

userName
String
the username of the user

password
String
the users password

2.8 PasswordForm

A form that collects user data needed when requesting a new password

The form contains the `userName` of the user requesting a new password.

When a new password is created, the password will be available in the `password`-property of the form.

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.PasswordForm"
  type="com.escenic.profile.presentation.struts.PasswordForm"/>
```

Action Error

This form requires that `userName` is set. If missing, the user should be notified.

no_userName
(On the `userName` property.) The `userName` was not specified

`PasswordForm` inherits properties from:

- `com.escenic.profile.presentation.struts.AbstractForm`

It also has the following properties of its own:

password
String
the new password of the user

userName

String

the username of the user

2.9 ValidateUserPasswordForm

A form that collects user data needed when requesting a new password

The form contains the `userName` and `validateField` of the user requesting a new password.

When a new password is created, the password will be available in the `password`-property of the form.

The form validates that user actually is a user and that both the username and emailaddress entered is the same as the one defined in the user itself before resetting the password. If this validation is not performed, it is possible for whoever to reset a password for whoever user.

If it's required to validate another field than the emailaddress, create a new subclass and override method `protected boolean isValidField(Person p)`

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.ValidateUserPasswordForm"
  type="com.escenic.profile.presentation.struts.ValidateUserPasswordForm"/>
```

Action Error

This form requires that `userName` and `validateField` is set. If missing, the user should be notified.

no_username

(On the `userName` property.) The `userName` was not specified

no_email

(On the `validateField` property.) The `validateField` was not specified

nameAndMailDoesNotMatch

(On the `global` property.) The `validateField` was not specified

`ValidateUserPasswordForm` inherits properties from:

- `com.escenic.profile.presentation.struts.PasswordForm`

It has no properties of its own.

2.10 ChangePasswordForm

A form that collects data needed when requesting a change of password

The form contains the old and new password

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.ChangePasswordForm"
  type="com.escenic.profile.presentation.struts.ChangePasswordForm"/>
```

Action Error

This form requires that both old and new password is set. If missing, the user should be notified.

```
no_Password
  (On the password property.) The new password was not specified

no_Old_Password
  (On the oldPassword property.) The old password was not specified
```

ChangePasswordForm inherits properties from:

- **com.escenic.profile.presentation.struts.AbstractForm**

It also has the following properties of its own:

```
password
String
  the new password of the user

oldPassword
String
  the oldPassword of the user
```

2.11 DeleteProfileForm

A form contains the name of a profile which is to be deleted

The properties describes below are all settable from the html form in the jsp page.

The form contains information about the profile to delete

Struts config

```
<form-bean
  name="com.escenic.profile.presentation.struts.DeleteProfileForm"
  type="com.escenic.profile.presentation.struts.DeleteProfileForm"/>
```

Action Error

This form requires that title is set. If missing, the user should be notified.

```
no_title
  (On the title property.) The title was not specified
```

DeleteProfileForm inherits properties from:

- **com.escenic.profile.presentation.struts.AbstractForm**

It has no properties of its own.

2.12 SearchForm

Struts config

```
<form-bean
  name="com.escenic.search.SearchForm"
  type="com.escenic.search.SearchForm"/>
```

Action Error

This are the errors returned by the validate method found on this action form.

errors.required

(On the **pageLength** property.) when a value is missing for this property

errors.integer

(On the **pageLength** property.) when the value of this property can't be converted to a integer

errors.minvalue

(On the **pageLength** property.) when the value of this property is less or equal to '0'

errors.required

(On the **pageNumber** property.) when a value is missing for this property

errors.integer

(On the **pageNumber** property.) when the value of this property can't be converted to a integer

errors.minvalue

(On the **pageNumber** property.) when the value of this property is smaller than '1'

errors.required

(On the **pageLength** property.) when a value is missing for this property

errors.integer

(On the **pageLength** property.) when the value of this property can't be converted to a integer

errors.minvalue

(On the **pageLength** property.) when the value of this property is less or equal to '0'

SearchForm inherits properties from:

- **com.escenic.tools.struts.ValidateForm**

It also has the following properties of its own:

pageLength

String

The number of search hits to show on each result page.; default value=10

pageNumber

String

the page in the search result we want to display.; default value=1

searchEngineName

String

The name of the search engine to use for search. When no name is given the default search engine configured by the system will be used.; default value='null'

publicationId

String

The publication to search; default value='null'

2.13 ArticleSearchForm

This is the abstract super class of all article search forms.

Struts config

```
<form-bean
  name="com.escenic.search.ArticleSearchForm"
  type="com.escenic.search.ArticleSearchForm"/>
```

Action Error

This are the errors returned by the validate method found on this action form.

errors.integer

(On the **includeSectionId** property.) when the value of this property can't be converted to a integer

errors.required

(On the **includeSubSections** property.) when a value is missing for this property

errors.integer

(On the **excludeSectionId** property.) when we are not able to produce a Integer.

errors.required

(On the **maxRows** property.) when a value is missing for this property

errors.integer

(On the **maxRows** property.) when the value of this property can't be converted to a integer

errors.required

(On the **optimized** property.) when a value is missing for this property

ArticleSearchForm inherits properties from:

- **com.escenic.search.SearchForm**

It also has the following properties of its own:

includeSectionId

String[]

The list over sections that is to be included. The list is made up of Strings that can be converted to Integers.

includeSubSections

String

Should we include sub sections in the search.; default value=false

excludeSectionId**String[]**

The list over sections that is to be excluded. The list is made up of Strings that can be converted to Integers.

articleType**String[]**

The Array of article types to be included in the search.; default value=an empty array, meaning that all article types is to be included

sortString**String**

The String used for deciding the sort order.; default value=an empty String.

sortStrings**List**

A list used to prepopulate the sortString properties. The list contains all legal values for the sortString property.

descending**String**

Should we sort decending or ascending.

maxRows**String**

the number of rows to return from the database.; default value=-1

searchType**String**

the search type

searchTypes**List**

A list used to prepopulate the searchType properties. The list contains all legal values for the searchType property.

optimized**String**

the optimized property

stemType**String**

the stem type property

stemTypes**List**

A list used to prepopulate the stemType properties. The list contains all legal values for the stemType property.

2.14 SimpleSearchForm

This is the search action form to use when you want to create a simple search.

Struts config

```
<form-bean
```

```
name="simpleSearchForm"  
type="com.escenic.search.SimpleSearchForm"/>
```

Action Error

This are the errors returned by the validate method found on this action form.

```
errors.required  
(On the searchString property.) when a value is missing for this property
```

SimpleSearchForm inherits properties from:

- **com.escenic.search.ArticleSearchForm**

It also has the following properties of its own:

```
searchString  
String  
the string used when searching
```

2.15 AdvancedSearchForm

This is the search action form to use when you want to create a advanced search page.

Struts config

```
<form-bean  
name="advancedSearchForm"  
type="com.escenic.search.AdvancedSearchForm"/>
```

Action Error

This are the errors returned by the validate method found on this action form.

```
errors.required  
(On the searchExpression property.) when a value is missing for this property
```

```
errors.required  
(On the startDay property.) when a value is missing for this property
```

```
errors.integer  
(On the startDay property.) when the value of this property can't be converted to a integer
```

```
errors.range  
(On the startDay property.) when the value of this property is not between '1' and '31'
```

```
errors.required  
(On the startMonth property.) when a value is missing for this property
```

```
errors.integer  
(On the startMonth property.) when the value of this property can't be converted to a integer
```

```
errors.range  
(On the startMonth property.) when the value of this property is not between '0' and '11'
```

errors.required

(On the **startYear** property.) when a value is missing for this property

errors.integer

(On the **startYear** property.) when the value of this property can't be converted to a integer

errors.range

(On the **startYear** property.) when the value of this property is not between '0' and '{@link Integer#MAX_VALUE}'

errors.date

(On the **fromDate** property.) when we are not able to produce a date.

errors.required

(On the **toDay** property.) when a value is missing for this property

errors.integer

(On the **toDay** property.) when the value of this property can't be converted to a integer

errors.range

(On the **toDay** property.) when the value of this property is not between '1' and '31'

errors.required

(On the **toMonth** property.) when a value is missing for this property

errors.integer

(On the **toMonth** property.) when the value of this property can't be converted to a integer

errors.range

(On the **toMonth** property.) when the value of this property is not between '0' and '11'

errors.required

(On the **toYear** property.) when a value is missing for this property

errors.integer

(On the **toYear** property.) when the value of this property can't be converted to a integer

errors.range

(On the **toYear** property.) when the value of this property is not between '0' and '11'

errors.date

(On the **toDate** property.) when we are not able to produce a date.

AdvancedSearchForm inherits properties from:

- **com.escenic.search.ArticleSearchForm**

It also has the following properties of its own:

all (read/write)**String**

All the words written here must be present in the article to get a hit.

exactPhrase (read/write)**String**

All words written here will be searched for as a phrase.

atLeastOne (read/write)**String**

At least one of these words must be present.

without (read/write)**String**

These words must NOT be present.

startDay (read/write)**String**

The day on which to start searching; default value=1

startMonth (read/write)**String**

The month on which to start searching; default value=0

startYear (read/write)**String**

The year on which to start searching; default value=current year

toDay (read/write)**String**

The day on which to stop searching; default value=1

toMonth (read/write)**String**

The month on which to stop searching; default value=0

toYear (read/write)**String**

The year on which to stop searching; default value=0

days (read-only)**List**

A list used to prepopulate the startDay and toDay properties. The list contains all values in the range from '1' to '31'. Both 'label' and 'value' is the number representation of the day of the month.

months (read-only)**List**

A list used to prepopulate the startMonth and toMonth properties. The list contains all values in the range from '0' to '11'. Both 'label' and 'value' is the number representation of the month of the year.

years (read-only)**List**

A list used to prepopulate the startYear and toYear properties. The list contains all values in the range from '1990' to 'currentYear'. Both 'label' and 'value' is the number representation of the year.

3 Actions

3.1 AbstractProfileAction

A base class which has behaviour and properties which are common to all actions

It requires an AbstractForm and is responsible for forwarding the request to the appropriate forward according to the result of the action.

If the action is a success, it forwards to the targetUrl, if defined, or success.

If the action fails, the request is forwarded to errorUrl, if defined, or input page

Struts config

```
<action path="/some/path/AbstractProfileAction"
  type="com.escenic.profile.presentation.struts.AbstractProfileAction"
  name=""
  input="/some/path/AbstractProfileAction-input.jsp"
  validate="true">
</action>
```

Action Forwards

3.2 AddUserAction

Creates a new user in escenic based on the values defined in the form. It creates both the user and profiles

Struts config

```
<action path="/newUser"
  type="com.escenic.profile.presentation.struts.AddUserAction"
  name="UserForm"
  input="/newUser-input.jsp"
  validate="true">

  <forward name="success"
    path="/newUser-success.jsp"/>
</action>
```

Action Form

This action can be used with the following form:

`com.escenic.profile.presentation.struts.AbstractUserForm`
The form containing the userdata

Action Forwards

input page

If the user couldn't be created.

success

If the user was successfully created

Action Error

If creating the user fails, an error will be logged, and the action will be forwarded to the input-page.

create_user_failed

(On the `global` property.) If creating user failes

3.3 LoginAction

Logs a user in to escenic

Struts config

```
<action path="/login"
  type="com.escenic.profile.presentation.struts.LoginAction"
  name="LogonForm"
  input="/login-input.jsp"
  validate="true">

  <forward name="success"
    path="/login-success.jsp"/>
</action>
```

Action Form

This action can be used with the following form:

com.escenic.profile.presentation.struts.LoginForm

The form contains information about the username and password to use when performing a login

Action Forwards

input page

If the user couldn't be logged in

success

If login was successfull.

Action Error

If login fails, an error will be logged and the action will be forwarded to the input-page and the user should be notified about what went wrong.

login_failed

(On the `global` property.) If logon fails

login_failed_other

(On the `global` property.) If logon fails due to unforeseen events

login_failed_account_expired

(On the `global` property.) If logon fails due to an expired account

3.4 NewPasswordAction

Resets the password of the given user

Struts config

```
<action path="/newPassword"
  type="com.escenic.profile.presentation.struts.NewPasswordAction"
  name="PasswordForm"
  input="/newPassword-input.jsp"
  validate="true">

  <forward name="success"
    path="/newPassword-success.jsp"/>
</action>
```

Action Form

This action can be used with the following form:

```
com.escenic.profile.presentation.struts.PasswordForm
```

The form contains information about the user which requires a new password

Action Forwards**input page**

If the password couldn't be reset

success

If reset was a success.

Action Error

If resetting password fails, an error will be logged, and the action will be forwarded to the input-page.

reset_password_failed

(On the `global` property.) If resetting password fails

3.5 SetPasswordAction

Resets the password of the current user. This can only be done in a logged-in context

Struts config

```
<action path="/setPassword"
  type="com.escenic.profile.presentation.struts.SetPasswordAction"
  name="ChangePasswordForm"
  input="/setPassword-input.jsp"
  validate="true">

  <forward name="success"
    path="/setPassword-success.jsp"/>
</action>
```

Action Form

This action can be used with the following form:

```
com.escenic.profile.presentation.struts.ChangePasswordForm
  The form contains information about the new password
```

Action Forwards

input page

If the password couldn't be updated

success

If update was a success.

Action Error

If updating password fails, an error will be logged, and the action will be forwarded to the input-page.

```
set_password_failed
  (On the global property.) If updating password failes
```

3.6 AuthenticatedAction

Some actions, such as for instance updating a users profile, should be done in context of a logged in user.

This action is an abstract superclass which ensures that a user is logged in before being able to perform the requested task. Subclasses needs to implement one method, `performAuthenticated`, to take advantage of user authentication.

The action ensures that the user is logged in before invoking the abstract method `performAuthenticated` which performs the requested action.

The abstract method `performAuthenticated` obeys the same contract as `Action#execute`.

Struts config

```
<action path="/doSomethingAuthenticated"
  type="com.escenic.profile.presentation.struts.AuthenticatedAction"
  name="SomeForm"
  input="/doSomethingAuthenticated-input.jsp"
  validate="true">

  <forward name="success"
    path="/doSomethingAuthenticated-success.jsp"/>
</action>
```

Action Forwards

input page

not-logged-in if the user was not logged in

success

If user is authenticated.

3.7 AuthenticatedForwardAction

This action ensures the user is logged in before forwarding the request to the success-mapping. If the user is not logged in, the user will be forwarded to "not-logged-in" mapping

Struts config

```
<action path="/doSomethingAuthenticated"
  type="com.escenic.profile.presentation.struts.AuthenticatedForwardAction"
  name="com.escenic.tools.struts.ForwardForm"
  input="/doSomethingAuthenticated-input.jsp"
  validate="true">

  <forward name="success"
    path="/doSomethingAuthenticated-success.jsp"/>
</action>
```

Action Forwards

input page

not-logged-in if the user was not logged in

success

If user is authenticated.

3.8 PopulateUserAction

Populates a form with data of the currently logged in user. This action does not lock the user when loaded.

Struts config

```
<action path="/populateUser"
  type="com.escenic.profile.presentation.struts.PopulateUserAction"
  name="UserForm"
  input="/populateUser-input.jsp"
  validate="true">

  <forward name="success"
    path="/populateUser-success.jsp"/>

  <forward name="not-logged-in"
    path="/populateUser-not-logged-in.jsp"/>
</action>
```

Action Form

This action can be used with the following form:

```
com.escenic.profile.presentation.struts.AbstractUserForm
  The form is populated with data from the user
```

Action Forwards

input page

If the user couldn't be loaded.

success

If the user was successfully loaded

not-logged-in

If user is not logged in

Action Error

If loading the user fails, an error will be logged, and the action will be forwarded to the input-page.

user_error

(On the `global` property.) If loading user failes failes

3.9 UpdateUserAction

Updates the logged in user with the values defined in the form. It both updates existing profiles and creates new profiles if user has added a new profile. The `userName` will not be updated using this action as this is a readonly value.

Struts config

```

<action path="/updateUser"
  type="com.escenic.profile.presentation.struts.UpdateUserAction"
  name="UserForm"
  input="/updateUser-input.jsp"
  validate="true">

  <forward name="success"
    path="/updateUser-success.jsp"/>

  <forward name="not-logged-in"
    path="/updateUser-not-logged-in.jsp"/>
</action>

```

Action Form

Below is a list of the Action Forms known to work with this Action. All of these Action Forms is a instance of {@link AbstractUserForm}.

DefaultUserForm

Implementation which uses the email as username.

StandardUserForm

Implementation which uses a separate field as username.

Action Forwards

input page

If the user couldn't be updated.

success

If the user was successfully updated

not-logged-in

If user is not logged in

Action Error

If updating the user fails, an error will be logged, and the action will be forwarded to the input-page.

update_user_failed

(On the `global` property.) If updating user failes

3.10 DeleteUserAction

Deletes the current user. The user will also be logged out of escenic as the user no longer has an available account.

Struts config

```

<action path="/deleteUser"
  type="com.escenic.profile.presentation.struts.DeleteUserAction"
  name=""
  input="/deleteUser-input.jsp"

```

```

validate="true">

<forward name="success"
  path="/deleteUser-success.jsp"/>

<forward name="not-logged-in"
  path="/deleteUser-not-logged-in.jsp"/>
</action>

```

Action Forwards

input page

If the user couldn't be deleted.

success

If the user was successfully deleted

not-logged-in

If user is not logged in

Action Error

If deleting the user fails, an error will be logged, and the action will be forwarded to the input-page.

could_not_delete_user

(On the `global` property.) If deleting user fails

3.11 DeleteProfileAction

Deletes a named profile from current user

Struts config

```

<action path="/deleteProfile"
  type="com.escenic.profile.presentation.struts.DeleteProfileAction"
  name=""
  input="/deleteProfile-input.jsp"
  validate="true">

  <forward name="success"
    path="/deleteProfile-success.jsp"/>

  <forward name="not-logged-in"
    path="/deleteProfile-not-logged-in.jsp"/>
</action>

```

Action Forwards

input page

If the profile couldn't be deleted.

success

If the profile was successfully deleted

not-logged-in

If user is not logged in

Action Error

If deleting the profile fails, an error will be logged, and the action will be forwarded to the error page.

couldNotFindProfile

(On the `global` property.) If the profile to delete could not be found

couldNotDeleteProfile

(On the `global` property.) If the profile couldn't be deleted

3.12 LogoffAction

Logs off currently logged in user. The session will still be valid, but the user will no longer be able to do authenticated actions

Struts config

```
<action path="/logoff"
  type="com.escenic.profile.presentation.struts.LogoffAction"
  name="LogoffForm"
  input="/logoff-input.jsp"
  validate="true">

  <forward name="success"
    path="/logoff-success.jsp"/>
</action>
```

Action Form

This action can be used with the following form:

```
com.escenic.profile.presentation.struts.LogoffForm
  Contains forwarding information
```

Action Forwards**success**

If logoff was a success

3.13 SearchAction

This is the component that does the actual search. The Search Action component will work with different Action forms. The only requirement is that all action forms are an instance of the Search Form class.

To provide a simple and an advanced search will you need to configure the SearchAction class twice in struts-config.xml. One for each form you will use.

Struts config

```
<action path="/search/simple"
  type="com.escenic.search.SearchAction"
  name="simpleSearchForm"
  input="/search/simple-input.jsp"
  validate="true">

  <forward name="success"
    path="/search/simple-success.jsp"/>
</action>
```

Action Form

Below is a list of the Action Forms known to work with this Action. All of these Action Forms is a instance of SearchForm.

com.escenic.search.SearchForm

This is the Abstract Super class for all SearchForms.

com.escenic.search.SimpleSearchForm

Use this Action Form when you want to create a simple search.

com.escenic.search.AdvancedSearchForm

Use this Action Form when you want to create a advanced search.

Action Forwards

This is a list of the action forwards used by this Action.

input page

When an error or something that requires the system to give the user a message, is the input parameter used. This forward can be overridden by providing a value for the 'input' property on the Action Form.

success

This forward is used when we have a result to display. Even a empty result is a result. This forward can be overridden by providing a value for the 'result' property on the Action Form.

Request attribute

The following request attributes are made available on the forwarded page.

result

This attribute is of type **com.escenic.search.ResultPage**. The list of hits for this search.

Action Error

This is a list of the errors that can be created by this action.

errors.form

(On the **ActionErrors.GLOBAL_ERROR** property.) When the Action Form provided is of the wrong type. {0} represents the provided Action Form, {1} the expected type.