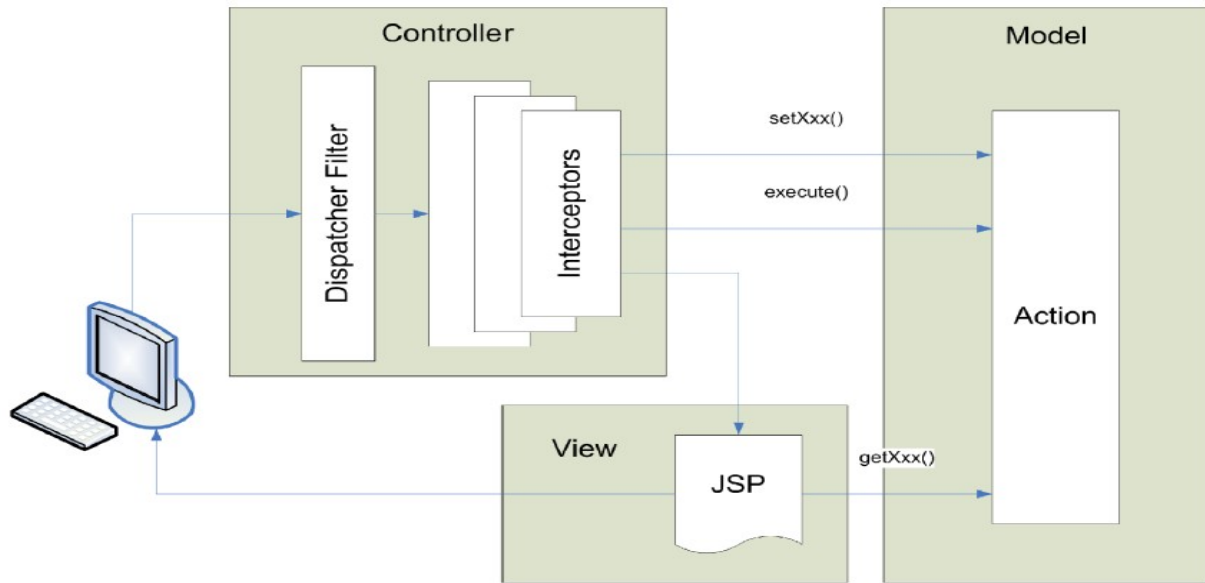


# Struts 2

## Série Java WEB – Cartão de Referência



### Configurando a aplicação WEB

O arquivo descritor **web.xml** contém um filtro e seu mapeamento. Por padrão, o filtro está mapeado para /\*, ou seja, todos os pedidos serão interceptados, mas apenas aqueles que termina com um sufixo específico (.action, por padrão) e alguns caminhos especiais (para arquivos estáticos) serão processados e tratados pelo Struts 2:

```
<web-app>
<...>
<filter>
  <filter-name>struts2</filter-name>
  <filter-class>
    org.apache.struts2.dispatcher.ng.filter.Struts
    PrepareAndExecuteFilter
  </filter-class>
</filter>
<filter-mapping>
  <filter-name>struts2</filter-name>
  <url-pattern>/*</url-pattern>
</filter-mapping>
<...>
</web-app>
```

### Configurando o Struts

Realizado no arquivo **struts.xml**, que contém as actions, validation, interceptors e results:

```
<struts>
<constant name="struts.custom.i18n.resources"
  value="global" />
<constant
  name="struts.enable.DynamicMethodInvocation"
  value="false" />
<constant name="struts.devMode"
  value="false" />

<package name="default" namespace="/"
  extends="struts-default">
  <default-action-ref name="menu" />
```

```
<global-results>
  <result name="error">/error.jsp</result>
</global-results>
<global-exception-mappings>
  <exception-mapping
    exception="java.lang.Exception"
    result="error"/>
</global-exception-mappings>
```

```
<!-- Ação Padrão -->
<action name="menu">
  <result>/pags/Menu.jsp</result>
</action>
```

```
<!-- Exemplo de um CRUD -->
<action name="lst[Tabela]"
  class="acao.Listagem[Tabela]"
  <result>/pags/[Tabela]Inc.jsp</result>
</action>
<action name="cad[Tabela]"
  class="acao.Modificar[Tabela]"
  method="inserir">
  <result name="sucesso" type="redirect">
    /lst[Tabela].action</result>
</action>
<action name="exc[Tabela]"
  class="acao.Modificar[Tabela]"
  method="excluir">
  <result name="sucesso" type="redirect">
    /lst[Tabela].action</result>
</action>
<action name="alt[Tabela]" method="obter"
  class="acao.Modificar[Tabela]"
  <result name="sucesso">
    /pags/[Tabela]Alt.jsp</result>
</action>
<action name="mod[Tabela]"
  class="acao.Modificar[Tabela]"
  method="alterar">
  <result name="sucesso" type="redirect">
    /lst[Tabela].action</result>
</action>
</package>
</struts>
```

# Struts 2

## Série Java WEB – Cartão de Referência

**DICA:** Planeje e elabore toda a estrutura do seu sistema, antes mesmo de começar a programar.

### Configurando a aplicação WEB

Nome da Tag	Descrição
constant	Modifica o valor da propriedade de configuração.
package	Providencia o caminho para a hierarquia, utilizado para separar a aplicação em unidades menores.
default-interceptor-ref	Este interceptor é utilizado para aplicar para todas as ações de pacote por padrão
global-results	Contém uma lista de resultado de tags.
global-exceptions-mappings	Contém uma lista de tags exception-mapping para aplicar as ações do pacote padrão.
exception-mapping (*)	Mapas exceção para o resultado que deve ser prestado quando a exceção é lançada.
default-action-ref	Esta ação é utilizada para chamar quando a URI não encontrar qualquer configuração
action	Descreve uma ação quando esta for chamada
result	Providencia uma opção de visão para o resultado que for retornado por uma ação.
interceptor-ref	Para aplicar a uma determinada ação.
param	Permite os parametros que devem assinalar a uma ação no arquivo de configuração
include	Inclui outros arquivos de configuração.

### Actions

No Struts 2, quando uma ação é executada, o resultado será utilizado para controlar o fluxo – Essas seqüências simples ("sucess", "error", "input", ..) será usado para chamar um determinado resultado - Este resultado pode ser um despachante para uma visão (arquivo JSP), renderizar um modelo Freemarker, gerar um gráfico, uma saída xml, entre outros, sendo totalmente independente de sua ação. Nota: quando a validação falhar, o resultado passado será por padrão "input".

#### Validação

01. Adicionar no struts.xml a tag **<result>** com a propriedade **name** igual a "input" para qual página deve ir quando a validação falhar.
02. Adicionar na tag **<s:form>** a propriedade **validate** com o valor igual a "true"
03. Na ação na qual este formulário chama adicionar o método com a assinatura "public void validate()", e caso falhe em algum teste, utilizar o método **addFieldError(cmp1,cmp2)**.

#### Tags de Controle

if            elseif        else            append        generator  
iterator     merge            sort            subset

#### Tags de Dados

a            action    bean        date        debug    i18n        include  
param     property   push        set        text        url

#### Tags do Struts

### Tags para os Formulários

checkbox   checkboxlist   combobox   doubleselect   head   file   form   hidden   label   radio  
optgroup   updownselect   password   optiontransferselect   select   submit   textarea   textfield   token   reset

```
<s:form action="crud!save" method="post">
<s:textfield name="employee.firstName" value="%{employee.firstName}"
  label="%{getText('label.firstName')}" size="40"/>
<s:textfield name="employee.lastName" value="%{employee.lastName}"
  label="%{getText('label.lastName')}" size="40"/>
<s:textfield name="employee.age" value="%{employee.age}" label="%{getText('label.age')}"
  size="20"/>
<s:select name="employee.department.departmentId" listValue="name"
  value="%{employee.department.departmentId}" list="departments" listKey="departmentId" />
<s:select name="gender" list="%{#{'male':'Male', 'female':'Female'}}" />
<s:select label="label" name="name" list="{ 'name1', 'name2', 'name3' }" value="%{'name2'}" />
<s:hidden name="employee.employeeId" value="%{employee.employeeId}"/>
<s:submit value="%{getText('button.label.submit')}" />
<s:submit value="%{getText('button.label.cancel')}" name="redirect-action:index"/>
</s:form>
```

### WEB-INF/classes/guest.properties

Este arquivo conterá chaves e valores para internacionalizar a aplicação WEB de uma forma simples. Por exemplo: ao invés de "Bem-vindo!" na sua página, deve-se usar especificamente uma chave (por exemplo 'WELCOME\_MSG') com um determinado valor ('Bem-vindo!'). Pela adição de novos pacotes de recursos, você pode substituir essa chave e especificar um valor diferente "Bienvenue!" por exemplo, para uma localidade francesa. **Tag genéricos** são utilizados para controlar o fluxo de execução quando as páginas render. Essas tags também permitem a extração de dados de outros lugares do que a sua ação ou a pilha de valor, tais como localização, JavaBeans, e incluindo URLs adicionais ou execuções ação.