

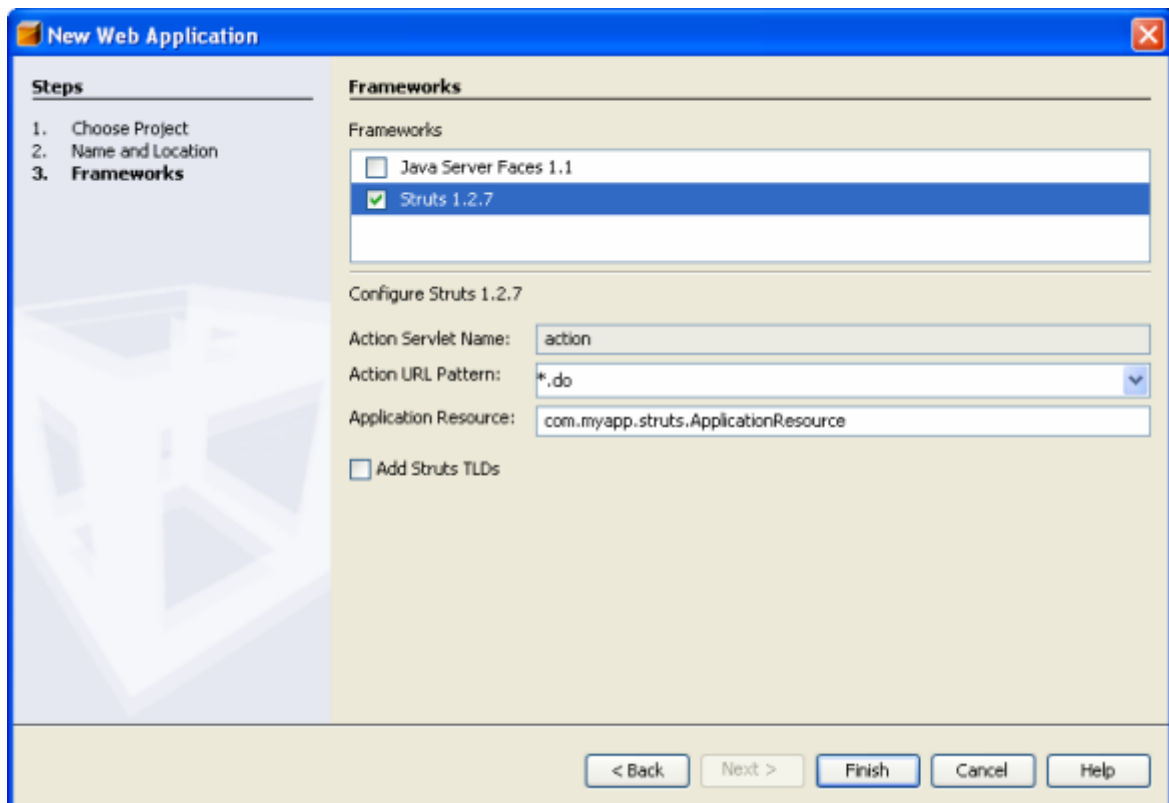
Configurando uma aplicação Struts

Instalando os Softwares

1. NetBeans IDE 5.0
2. Java Standard Development Kit (JDK™) version 1.4.2 or 5.0

Criando uma nova aplicação Struts

1. Click File > New Project. Em Categories, selecione Web. Em Projects, selecione Web Application e click em Next.
2. No Nome e Localidade faça o seguinte :
3. Em Project Name, entre com PaginaLogin.
4. Altere o diretorio do Proejto para \$PROJECTHOME.
5. Selecione a estrutura de arquivos
 - a. Java BluePrints.
<http://java.sun.com/blueprints/code/projectconventions.html>.
 - b. Jakarta. **<http://jakarta.apache.org/tomcat/tomcat-5.0-doc/>.**
6. Selecione o servidor para realizar o deploy da sua aplicação
7. Altere o Context Path para / PaginaLogin.
8. Click em Next.
9. Em Frameworks panel, selecione Struts 1.2.7:



Desenvolvendo uma aplicação Struts

Utilizando Taglib do Struts

1. Click botão direito no projeto selecione New > JSP, e escolha nova pagina JSP loginForm. Click em Finish.
2. Em Source Editor, altere o conteudo original na tag <H1> para Login Form e faça o mesmo na tag <TITLE> .
3. Copie as duas taglib do arquivo welcomeStruts.jsp para o novo arquivo loginForm.jsp:

```
<%@ taglib uri="http://jakarta.apache.org/struts/tags-bean"
prefix="bean" %>
<%@ taglib uri="http://jakarta.apache.org/struts/tags-html"
prefix="html" %>
```

4. em loginForm.jsp, abaixo da tag <H1> adicione o seguinte codigo :

```
<html:form action="login">
    <html:submit value="Login" />
</html:form>
```

5. Insira uma tabela antes de <html:submit value="Login" /> no arquivo loginForm.jsp

```
<html:form action="login">
    <table border="1">
        <thead>
            <tr>
                <th></th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td></td>
            </tr>
        </tbody>
    </table>

    <html:submit value="Login" />
</html:form>
```

6. Entre a tag <TH> adicione o seguinte codigo:

```
<bean:message key="login.name" />
```

7. Entre a tag <TD> adicione o seguinte codigo:

```
<html:text property="name" />
```

8. No arquivo ApplicationResource.properties adicione a chave login.name e um valor conforme exemplo:

```
login.name=Name
```

Utilizando Validate do Struts

1. Click botão direito no projeto selecione New > File/Folder. Na categoria Web escolha Struts ActionForm Bean e click em Next. Selecione o pacote `com.myapp.struts` e click em Finish.
2. Abra o arquivo `struts-config.xml` no editor e veja o seguinte código :

```
<form-beans>
    <form-bean name="NewStrutsActionForm"
type="com.myapp.struts.NewStrutsActionForm"/>
</form-beans>
```

3. Veja o método `validate` do `form-beans` conforme o seguinte código :

```
public ActionErrors validate(ActionMapping mapping, HttpServletRequest
request) {
    ActionErrors errors = new ActionErrors();
    if (getName() == null || getName().length() < 1) {
        errors.add("name", new
ActionMessage("error.name.required"));
        // TODO: add 'error.name.required' key to your resources
    }
    return errors;
}
```

4. Adicione a chave `error.name.required` no arquivo `ApplicationResource.properties` conforme o exemplo:

```
error.name.required=Enter a name!
```

adicione no topo do arquivo `ApplicationResource.properties` as chaves de configurações abaixo:

```
errors.header=
errors.prefix=<span style="color: red">
errors.suffix=</span>
errors.footer=
```

5. adicione a tab de erros no arquivo `loginForm.jsp`, depois da tag `</html:form>`:

```
<html:errors />
```

Utilizando taglib do Struts

1. Click botão direito no projeto selecione New > File/Folder. Na categoria Web escolha Struts Action e click em Next.
2. Crie a classe "NewStrutsAction".
3. Selecione o pacote `com.myapp.struts`.
4. Digite `login` em Action Path . Click em Next.
5. Selecione o `NewStrutsActionForm` como ActionForm Bean, Em Input Resource digite `loginForm.jsp` . Click em Finish.
6. Abra `struts-config.xml` e veja o código :

```
<action-mappings>
    <action input="/loginForm.jsp"
        name="NewStrutsActionForm"
```

```

        path="/login"
        scope="session"
        type="com.myapp.struts.NewStrutsAction"/>
<action path="/Welcome"
        forward="/welcomeStruts.jsp"/>
</action-mappings>

```

7. Veja o metodo execute :

```

public ActionForward execute(ActionMapping mapping, ActionForm form,
    HttpServletRequest request, HttpServletResponse response)
    throws Exception {

    return mapping.findForward(SUCCESS);
}

```

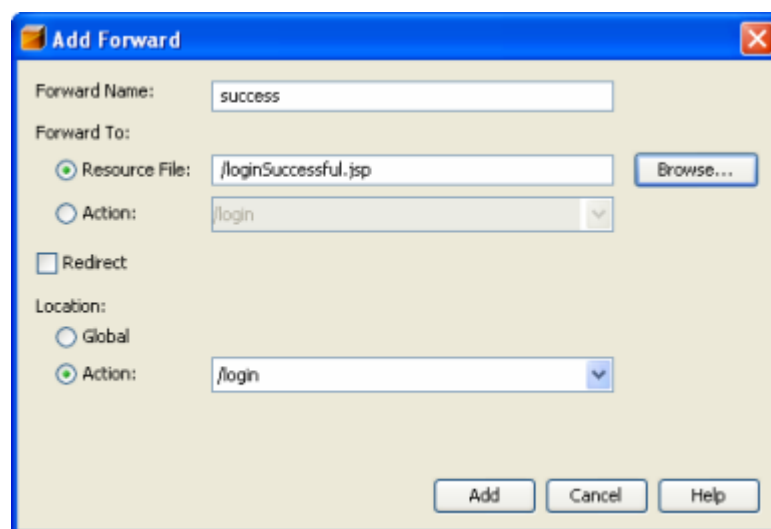
Veja a constante SUCCESS :

```

private final static String SUCCESS = "success";

```

8. Criar a o JSP login form com o texto " Login com Sucesso "
9. Veja o arquivo struts-config.xml e click com o botão direito em /login action mapping, e selecione Struts > Add Forward.



Veja o arquivo struts-config.xml abaixo como deve ficar :

```

<action-mappings>

    <action input="/loginForm.jsp"
        name="NewStrutsActionForm"
        path="/login"
        scope="session"
        type="com.myapp.struts.NewStrutsAction">
        <forward name="success"
            path="/loginSuccessful.jsp"/>
    </action>

    <action path="/Welcome"
        forward="/welcomeStruts.jsp"/>

</action-mappings>

```

Rodando a aplicação Struts

1. Click botão direito no projeto selecione Properties. Em Run defina a URL para login.do Click OK.
2. Escolha Run > Run Main Project (F6).

Se clicar em login ira ser redirecionado para loginSucessful.jsp

Adicionando funcionalidades na aplicação Struts

Adicionando a funcionalidade cancelar

1. Em loginForm.jsp, depois da tag <html:submit> crie o botão cancelar conforme exemplo :

```
<html:cancel />
```

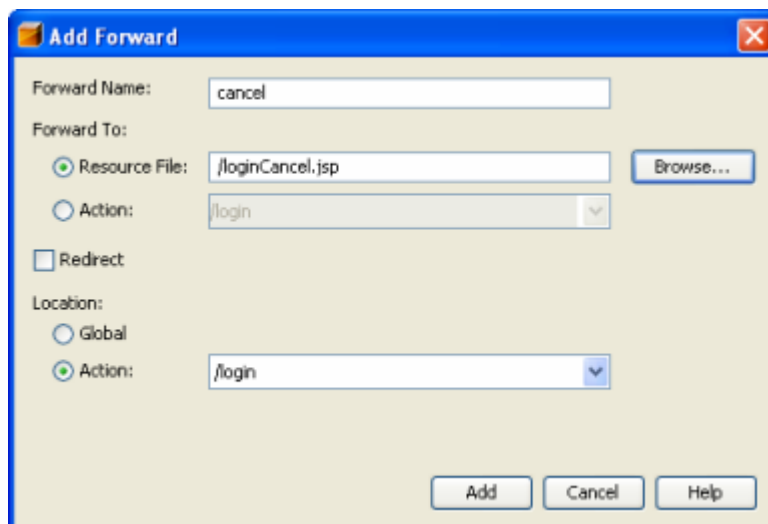
2. Adicione o seguinte codigo no metodo execute da classe org.myapp.struts.NewStrutsAction:

```
if (isCancelled(request)){  
    return mapping.findForward(CANCEL);  
}
```

Adicione as constante CANCEL

```
private final static String CANCEL = "cancel";
```

3. Crie a pagina loginCancel.jsp com o texto de login cancelado.
4. Abra o arquivo struts-config.xml em /login action mapping, e escolha Struts > Add Forward.



Veja o arquivo struts-config.xml abaixo como deve ficar :

```
<action-mappings>  
  
    <action input="/loginForm.jsp"
```

```

        name="NewStrutsActionForm"
        path="/login"
        scope="session"
        type="com.myapp.struts.NewStrutsAction">
        <forward name="success"
            path="/loginSuccessful.jsp"/>
        <forward name="cancel"
            path="/loginCancel.jsp"/>
    </action>

    <action path="/Welcome"
        forward="/welcomeStruts.jsp"/>

</action-mappings>

```

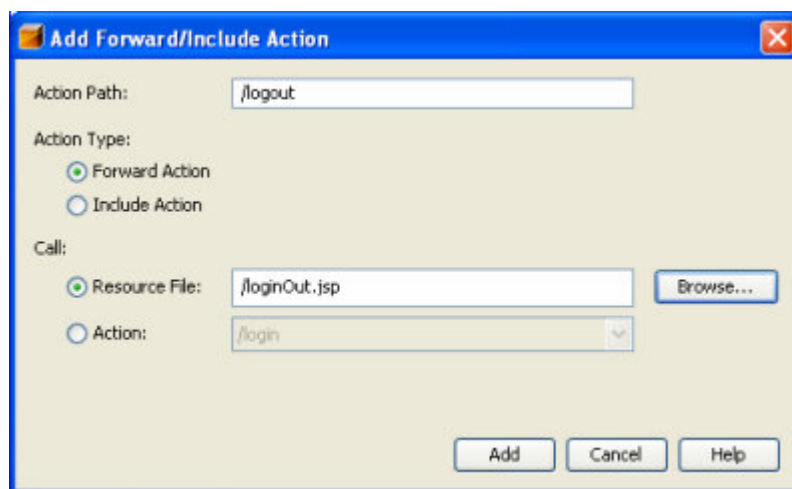
5. Rode a aplicação e click em cancel :

Adicionando a funcionalidade Logout

1. no arquivo loginForm.jsp, adicione o depois da tag <H1> o seguinte código:

```
<html:link action="/logout">Logout</html:link>
```

2. Crie a pagina JSP loginOut.jsp com o texto Tenha um otimo dia.
3. Abra o arquivo struts-config.xml e escolha Struts > Add Forward/Include Action.



Veja o arquivo struts-config.xml abaixo como deve ficar :

```

<action-mappings>

    <action input="/loginForm.jsp"
        name="NewStrutsActionForm"
        path="/login"
        scope="session"
        type="com.myapp.struts.NewStrutsAction">
        <forward name="success"
            path="/loginSuccessful.jsp"/>
        <forward name="cancel"
            path="/loginCancel.jsp"/>
    </action>

    <action forward="/loginOut.jsp"

```

```
        path="/logout"/>

        <action path="/Welcome"
                forward="/welcomeStruts.jsp"/>

    </action-mappings>
```

6. Rode a aplicação e click no link Logout:

Adicionando a funcionalidade Reset

1. Em `loginForm.jsp`, entre o botão Submit e Cancel adicione o seguinte código:

```
<html:reset />
```

2. Rode a aplicação e click no botão reset: