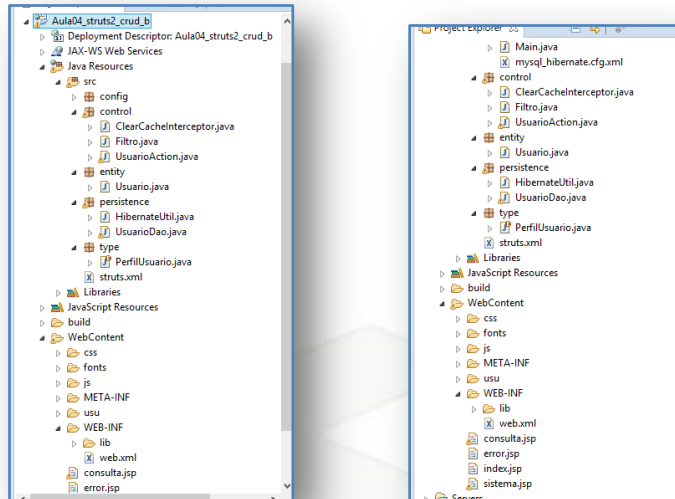


Estrutura do projeto...



LIBS:

antlr-2.7.6.jar
asm-attrs.jar
asm.jar
c3p0-0.9.0.jar
cglib-2.1.3.jar
commons-beanutils-1.8.0.jar
commons-codec-1.3.jar
commons-collections-3.2.1.jar
commons-digester-2.0.jar
commons-el-1.0.jar
commons-fileupload-1.0.jar
commons-lang-2.1.jar
commons-logging-1.1.1.jar
commons-validator.jar
dom4j-1.6.1.jar
ejb3-persistence.jar
freemarker-2.3.8.jar
hibernate-annotations.jar
hibernate-commons-annotations.jar
hibernate3.jar
jakarta-oro.jar
jcommon_javadoc.jar

```
jcommon_source.jar
jcommon.jar
jstl.jar
jta.jar
log4j-1.2.15.jar
mysql-connector-java-5.1.23-bin.jar
ognl-2.6.11.jar
oro-2.0.8.jar
servlet3-api.jar
standard.jar
struts2-core-2.0.14.jar
xwork-2.0.7.jar
```

```
package entity;
```

```
import java.io.Serializable;
```

```
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;
```

```
@Entity
@Table
@NamedQueries({
    @NamedQuery(name="Usuario.findAll", query="from Usuario"),
    @NamedQuery(name="Usuario.findByLogin",
        query="from Usuario u where u.email =:param1 and u.senha
        =:param2")
})
```

```
public class Usuario implements
Serializable{
```

```
    private static final long serialVersionUID = 1L;
```

```
    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)
    @Column
```

```
private Integer idUsuario;

@Column(unique=true, length=50, nullable=false)
private String email;

@Column(length=50, nullable=false)
private String nome;

@Column( length=100, nullable=false)
private String senha;

@Column( length=10, nullable=false)
private String perfil;

public Usuario() {
}

public Usuario(Integer idUsuario, String email, String
nome, String senha, String perfil) {
    this.idUsuario = idUsuario;
    this.email = email;
    this.nome = nome;
    this.senha = senha;
    this.perfil = perfil;
}

@Override
public String toString() {
    return "Usuario [idUsuario=" + idUsuario + ", email="
+ email + ", nome=" + nome + ", perfil=" + perfil + "];"
}

public Integer getIdUsuario() {
    return idUsuario;
}

public void setIdUsuario(Integer idUsuario) {
    this.idUsuario = idUsuario;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
    this.email = email;
}

public String getNome() {
```

```
        return nome;
    }
    public void setName(String nome) {
        this.nome = nome;
    }
    public String getSenha() {
        return senha;
    }
    public void setSenha(String senha) {
        this.senha = senha;
    }
    public String getPerfil() {
        return perfil;
    }
    public void setPerfil(String perfil) {
        this.perfil = perfil;
    }
    public static long getSerialVersionUID() {
        return serialVersionUID;
    }
}
```

mysql_hibernate.cfg.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate
Configuration DTD 3.0//EN"
"http://hibernate.sourceforge.net/hibernate-configuration-
3.0.dtd">
<hibernate-configuration>

    <session-factory>
        <property
name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</pro
perty>
        <property
name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</
property>
        <property name="hibernate.connection.url">
jdbc:mysql://localhost:3306/struts2?createDatabaseIfNotExist=tru
e</property>
        <property
name="hibernate.connection.username">root</property>
```

```
<property
name="hibernate.connection.password">coti</property>

<property name="hibernate.show_sql">true</property>
<property name="hibernate.format_sql">true</property>
<property name="hibernate.hbm2ddl.auto">update</property>

<mapping class="entity.Usuario"/>

</session-factory>
</hibernate-configuration>
```

```
package config;

import org.hibernate.cfg.AnnotationConfiguration;
import org.hibernate.cfg.Configuration;
import org.hibernate.tool.hbm2ddl.SchemaExport;

public class Main {

    public static void main(String[] args) {
        Configuration cfg = new AnnotationConfiguration().
configure("config/mysql_hibernate.cfg.xml");
        new SchemaExport(cfg).create(true, true);
        System.out.println("Tabela Gerada com Sucesso ...");
    }
}
```

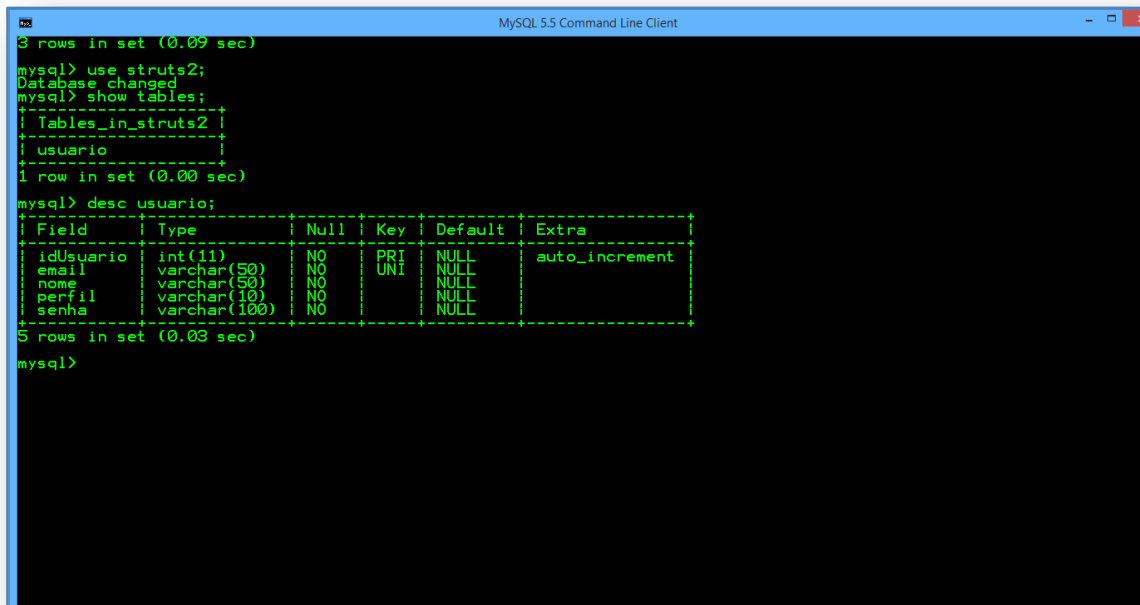
No console...

```
log4j:WARN No appenders could be found for logger
(org.hibernate.cfg.annotations.Version).
log4j:WARN Please initialize the log4j system properly.
```

```
drop table if exists Usuario
```

```
create table Usuario (
    idUsuario integer not null auto_increment,
    email varchar(50) not null unique,
    nome varchar(50) not null,
    perfil varchar(10) not null,
    senha varchar(100) not null,
```

)
primary key (idUserario)
Tabela Gerada com Sucesso ...



```
mysql> use struts2;
Database changed
mysql> show tables;
+-----+
| Tables_in_struts2 |
+-----+
| usuario           |
+-----+
1 row in set (0.00 sec)

mysql> desc usuario;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key  | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| idUsuario  | int(11)       | NO   | PRI  | NULL    | auto_increment |
| email      | varchar(50)   | NO   | UNI  | NULL    |                |
| nome       | varchar(50)   | NO   |      | NULL    |                |
| perfil     | varchar(10)   | NO   |      | NULL    |                |
| senha      | varchar(100)  | NO   |      | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.03 sec)

mysql>
```

`package` type;

`//Classe Estatica(Constante)`

`public enum PerfilUsuario {`

`ADMINISTRADOR("adm"),USUARIO("usu"),GUEST("guest");`

`private String perfil;`

`PerfilUsuario(String perfil){`
`this.perfil = perfil;`
`}`

`public String getPerfil() {`
`return perfil;`
`}`

`}`

```
package control;
```

```
import javax.servlet.http.HttpServletResponse;
```

```
import org.apache.struts2.ServletActionContext;
```

```
import com.opensymphony.xwork2.ActionInvocation;
```

```
import com.opensymphony.xwork2.interceptor.Interceptor;
```

```
public class ClearCacheInterceptor  
implements Interceptor{
```

```
    private static final long serialVersionUID = 1L;
```

```
    @Override
```

```
    public void destroy() {
```

```
        // TODO Auto-generated method stub
```

```
    }
```

```
    @Override
```

```
    public void init() {
```

```
        // TODO Auto-generated method stub
```

```
    }
```

```
    @Override
```

```
    public String intercept(ActionInvocation invocation) throws  
Exception {
```

```
        HttpServletResponse response =  
ServletActionContext.getResponse();
```

```
System.out.println("Chegou no Intercept -----");
```

```
        response.setHeader("Cache-Control", "no-cache, no-store");
```

```
        response.setHeader("Pragma", "no-cache");
```

```
        response.setDateHeader("Expires", 0);
```

```
        return invocation.invoke();
```

```
    }
```

```
}
```

```
package control;
```

```
import java.io.IOException;
```

```
import javax.servlet.Filter;
```

```
import javax.servlet.FilterChain;
```

```
import javax.servlet.FilterConfig;
```

```
import javax.servlet.ServletException;
import javax.servlet.ServletRequest;
import javax.servlet.ServletResponse;
import javax.servlet.annotation.WebFilter;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

@WebFilter({"/usu/*", "/adm/*"})
public class Filtro implements Filter
{

    public Filtro() {
    }

    public void destroy() {
    }

    public void doFilter(ServletRequest request,
        ServletResponse response, FilterChain chain) throws IOException,
        ServletException {
        HttpServletRequest req= (HttpServletRequest) request;
        HttpServletResponse resp = (HttpServletResponse)
response;
        HttpSession session = req.getSession();
        if (session.getAttribute("usuario")==null){
            resp.sendRedirect("../index.jsp");
//va para o index.jsp
            return; //para
        }
//senão dá sequencia ...
        chain.doFilter(request, response);
    }

    public void init(FilterConfig fConfig) throws
        ServletException {
    }
}
```



```
package control;

import java.util.ArrayList;
import java.util.List;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.struts2.ServletActionContext;

import persistence.UsuarioDao;

import com.opensymphony.xwork2.ActionContext;
import com.opensymphony.xwork2.ActionSupport;
import com.opensymphony.xwork2.ModelDriven;

import entity.Usuario;

public class UsuarioAction extends
ActionSupport
implements ModelDriven<Usuario>{

    private static final long serialVersionUID = 1L;

    //Nome do Objeto
    private Usuario usuario; //Classe Usuario
    private List <Usuario> lista;

    HttpServletRequest request;
    HttpSession session; //Irei trabalhar com Sessão

    public UsuarioAction() {
        usuario = new Usuario(); //espaço no Construtor
        lista =new ArrayList<Usuario>(); //construtor
    }

    @Override
    public Usuario getModel() {
        return usuario;
    }
    public Usuario getUsuario() {
        return usuario;
    }
}
```

```
}  
public void setUsuario(Usuario usuario) {  
    this.usuario = usuario;  
}  
public static long getSerialVersionUID() {  
    return serialVersionUID;  
}  
}  
  
public String listar(){  
    return SUCCESS;  
}  
public List<Usuario> getLista() {  
    try{  
        lista = new UsuarioDao().findAll();  
//chamada do Banco  
    }catch(Exception ex){  
    }  
    return lista;  
}  
  
public void setLista(List<Usuario> lista) {  
    this.lista = lista;  
}  
  
public String gravar(){  
    request = (HttpServletRequest)  
    ActionContext.getContext().get  
    (ServletActionContext.HTTP_REQUEST);  
    try{  
        new UsuarioDao().create(usuario);  
        getLista();  
        usuario = new Usuario();  
        request.setAttribute("msg", "Dados Gravados");  
    }catch(Exception ex){  
        request.setAttribute("msg", "Error :" + ex.getMessage());  
    }  
    return SUCCESS;  
}  
  
public String login(){  
    request = (HttpServletRequest)  
    ActionContext.getContext().get  
    (ServletActionContext.HTTP_REQUEST);
```

```
        session = request.getSession(true);
        try{
            Usuario resp = new UsuarioDao().findByLogin(usuario);
            if (resp!=null){
                usuario = new Usuario();
                session.setAttribute("usuario", resp);
            session.setAttribute("msg","Logado Com Sucesso");
            return SUCCESS;
            } else{
            session.setAttribute("usuario", null); //limpa
            }
            request.setAttribute("msg", "Usuario NAO LOGADO");
            return INPUT; //sistema.jsp
        }catch(Exception ex){
            request.setAttribute("msg", "Error :" + ex.getMessage());
            return ERROR; //erro.jsp
        }
    }

    public String logout(){
        request = (HttpServletRequest)
        ActionContext.getContext().get(ServletActionContext.HTTP_REQUEST
        );

        session = request.getSession(true);
        try{
            session.setAttribute("msg", null);
            session.setAttribute("usuario", null);
            session.invalidate();
            return SUCCESS;
        }catch(Exception ex){
            request.setAttribute("msg","Error:" + ex.getMessage());
        }
        return ERROR;
    }
}
```

```
package persistence;
```

```
import org.hibernate.cfg.AnnotationConfiguration;  
import org.hibernate.SessionFactory;
```

```
public class HibernateUtil {  
    private static final SessionFactory sessionFactory;  
  
    static {  
        try {  
            sessionFactory = new AnnotationConfiguration().  
configure("config/mysql_hibernate.cfg.xml").buildSessionFactory(  
);  
        } catch (Throwable ex) {  
            System.err.println("Initial SessionFactory creation  
failed." + ex);  
            throw new ExceptionInInitializerError(ex);  
        }  
    }  
  
    public static SessionFactory getSessionFactory() {  
        return sessionFactory;  
    }  
}
```

```
package persistence;
```

```
import java.util.List;
```

```
import org.hibernate.Criteria;  
import org.hibernate.Query;  
import org.hibernate.Session;  
import org.hibernate.Transaction;  
import org.jcommon.encryption.SimpleMD5;
```

```
import entity.Usuario;
```

```
public class UsuarioDao {
```

```
    Session session;  
    Transaction transaction;  
    Query query;  
    Criteria criteria;
```

```
private void criptografia(Usuario u){
    SimpleMD5 md5 = new
SimpleMD5(u.getSenha(), "cotiinformatica@gmail");
    u.setSenha(md5.toHexString());
}

public void create(Usuario u){
    criptografia(u);
    session = HibernateUtil.getSessionFactory().openSession();
    transaction = session.beginTransaction();
    session.save(u);
    transaction.commit();
    session.close();
}

public Usuario findByLogin(Usuario u){
    criptografia(u);
    session = HibernateUtil.getSessionFactory().openSession();
    query = session.getNamedQuery("Usuario.findByLogin");
    query.setString("param1", u.getEmail());
    query.setString("param2", u.getSenha());
    Usuario resp=(Usuario) query.uniqueResult();
    session.close();
    return resp;
}

public List<Usuario> findAll(){
    session = HibernateUtil.getSessionFactory().openSession();
    query = session.getNamedQuery("Usuario.findAll");
    List<Usuario> lista =query.list();
    session.close();
    return lista;
}
}
```

struts.xml

```
<!DOCTYPE struts PUBLIC
"-//Apache Software Foundation//DTD Struts Configuration
2.0//EN"
"http://struts.apache.org/dtds/struts-2.0.dtd">

<struts>
  <package name="usuarioaction" extends="struts-default">
    <interceptors>
      <interceptor name="caching"
class="control.ClearCacheInterceptor"/>
      <interceptor-stack name="cachingStack">
        <interceptor-ref name="caching" />
        <interceptor-ref name="defaultStack" />
        <interceptor-ref name="token"/>
      </interceptor-stack>
    </interceptors>

    <action name="usuario.gravar"
      class="control.UsuarioAction"
      method="gravar">
      <result name="success"/>/sistema.jsp</result>
      <result name="input"/>/sistema.jsp</result>
      <result name="error"/>/error.jsp</result>
    </action>

    <action name="usuario.listar"
      class="control.UsuarioAction"
      method="listar">
      <result name="success"/>/consulta.jsp</result>
    </action>

    <action name="usuario.Logar"
      class="control.UsuarioAction"
      method="Logar">
      <interceptor-ref name="cachingStack"/>
      <result name="success"/>/usu/logado.jsp</result>
      <result name="input"/>/sistema.jsp</result>
      <result name="error"/>/error.jsp</result>
      <result name="invalid.token"/>/sistema.jsp</result>
    </action>

    <action name="usuario.Logout"
```

```
        class="control.UsuarioAction"  
        method="Logout">  
        <result name="input">/usu/logado.jsp</result>  
        <result name="success">/sistema.jsp</result>  
        <result name="error">/error.jsp</result>  
    </action>  
</package>  
  
</struts>
```

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>  
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
xmlns="http://xmlns.jcp.org/xml/ns/javaee"  
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee  
http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd"  
id="WebApp_ID" version="3.1">  
  
    <filter>  
        <filter-name>struts2</filter-name>  
        <filter-  
class>org.apache.struts2.dispatcher.FilterDispatcher</filter-  
class>  
    </filter>  
  
    <filter-mapping>  
        <filter-name>struts2</filter-name>  
        <url-pattern>/*</url-pattern>  
    </filter-mapping>  
  
    <session-config>  
        <session-timeout>30</session-timeout>  
    </session-config>  
  
    <welcome-file-list>  
        <welcome-file>sistema.jsp</welcome-file>  
    </welcome-file-list>  
</web-app>
```

index.jsp

```
<!doctype html>
<html>

<meta http-equiv="Pragma" content="no-cache">
<meta http-equiv="Cache-Control" content="no-cache">
<meta http-equiv="Expires" content="0">
<%
    try {
        session.setAttribute("msg", null);
        session.invalidate();
    } catch (Exception ex) {
    }
%>
<head>
<link rel="stylesheet" href="css/bootstrap.css">
<script type="text/javascript" src="js/jquery-1.10.2.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
</head>

<body>
    <div class="container">
        <div class="well">
            <h2>Bem Vindo ao Sistema Struts2</h2>
            <a href="sistema.jsp" class="btn btn-primary">Entrar No Sistema <span class="glyphicon glyphicon-Login"></span></a>
        </div>

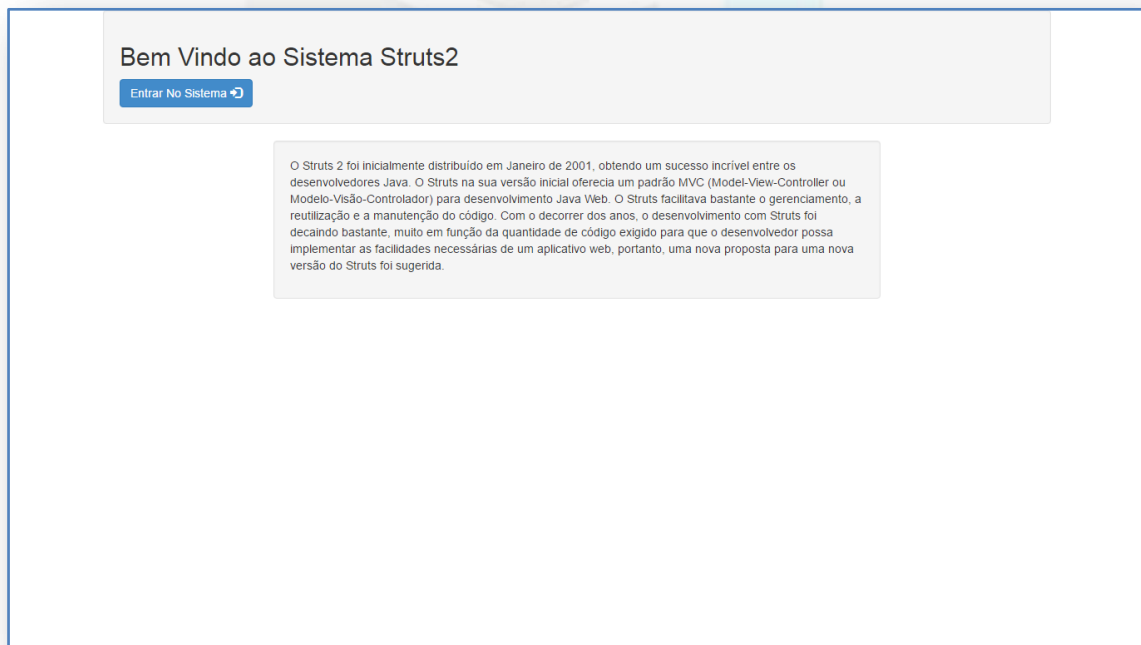
        <div class="col-sm-8 col-sm-offset-2">
            <div class="well">
```

O Struts 2 foi inicialmente distribuído em Janeiro de 2001, obtendo um sucesso incrível entre os desenvolvedores Java. O Struts na sua versão inicial oferecia um padrão MVC (Model-View-Controller ou Modelo-Visão-Controlador) para desenvolvimento Java Web. O Struts facilitava bastante o gerenciamento, a reutilização e a manutenção do código.

Com o decorrer dos anos, o desenvolvimento com Struts foi decaindo bastante, muito em função da quantidade de código exigido para que o desenvolvedor possa implementar as facilidades necessárias de um aplicativo web, portanto, uma nova proposta para uma nova versão do Struts foi sugerida.</p></div>

```
</div>
</div>

</div>
</body>
</html>
```



systema.jsp

```
<!DOCTYPE html>
<html>

<head>
<%@ taglib prefix="s" uri="/struts-tags"%>
<link rel="stylesheet" href="css/bootstrap.css">
```

```

<script type="text/javascript" src="js/jquery-
1.10.2.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
</head>

<body>
  <div class="container">
    <div class="well">
      <h2>Cadastro de Usuário em Struts2</h2>
      <s:a href="usuario.home.action" cssClass="btn btn-success">Home
      <span class="glyphicon glyphicon-home"></span> </s:a>
      <s:a href="usuario.listar.action" cssClass="btn btn-
warning">Consultar os Dados
      <span class="glyphicon glyphicon-search"></span></s:a>
    </div><!-- FECHA WELL -->

    <div class="col-sm-6 col-sm-offset-3">
      <ul class="nav nav-tabs">
        <li class="active"><a data-toggle="tab"
href="#home">Cadastro</a></li>
        <li><a data-toggle="tab"
href="#Logar">Logar </a></li>
      </ul>

      <div class="tab-content">
        <div id="home" class="tab-pane fade in active">
          <s:form method="post" action="usuario.gravar">
            <br />
            <input type="hidden" name="usuario.perfil" value="usu">

            <label for="nome">Nome :</label>
            <input type="text" name="usuario.nome" id="nome"
required="required" placeholder="Entre com Nome"
title="Entre com o Padrao no Nome" pattern="[A-Z
a-z]{3,35}"
class="form-control" />

            <label for="email">Email :</label>
            <input type="email" name="usuario.email"
required="required" placeholder="Entre com
Email"
title="Entre com o Padrao no Email" class="form-
control" />

            <label for="Senha">Senha :</label>
            <input type="password" name="usuario.senha"
required="required" placeholder="Entre com

```

```
Senha" title="Entre com o Padrao no Senha" class="form-
control" />
<br />
<input type="submit" value="Cadastrar Usuario"
class="btn btn-primary" />
</s:form>
</div><!-- FECHA DIV HOME -->

<!-- Até aqui cadastro, em baixo logar -->

<div id="Logar" class="tab-pane fade">
<s:form method="post" action="usuario.Logar"><br/>

<label for="email2">Email :</label>
<input type="email" name="usuario.email" id="email2"
required="required" placeholder="Entre com
Email" title="Entre com o Padrao no Email" class="form-
control" />

<label for="senha2">Senha :</label>
<input type="password" name="usuario.senha"
required="required" placeholder="Entre com
Senha" title="Entre com o Padrao no Senha" class="form-
control" />

<br />

<input type="submit" value="Logar" class="btn btn-primary" />
<s:token />
</s:form>
</div><!-- FECHA DIV LOGAR -->
</div><!-- FECHA TAB CONTENT -->
<br/>

<div class="panel panel-footer">
<footer>Java WebDeveloper -
www.cotiinformatica.com.br</footer>
</div><!-- FECHA PANEL FOOTER -->

    ${msg}
</div><!-- FECHA COL -->
</div> <!-- FECHA CONTAINER -->
</body>

</html>
```

Cadastro de Usuário em Struts2

[Home](#) [Consultar os Dados](#)

[Cadastro](#) [Logar](#)

Nome :

Email :

Senha :

[Cadastrar Usuario](#)

Java WebDeveloper - www.cotiinformatica.com.br

Cadastrando...

Cadastro de Usuário em Struts2

[Home](#) [Consultar os Dados](#)

[Cadastro](#) [Logar](#)

Nome :

Email :

Senha :

[Cadastrar Usuario](#)

Java WebDeveloper - www.cotiinformatica.com.br

Cadastrado...

Cadastro de Usuário em Struts2

[Home](#) [Consultar os Dados](#)

Nome :

Email :

Senha :

Java WebDeveloper - www.cotiinformatica.com.br

Dados Gravados

Aba de Logar...

Cadastro de Usuário em Struts2

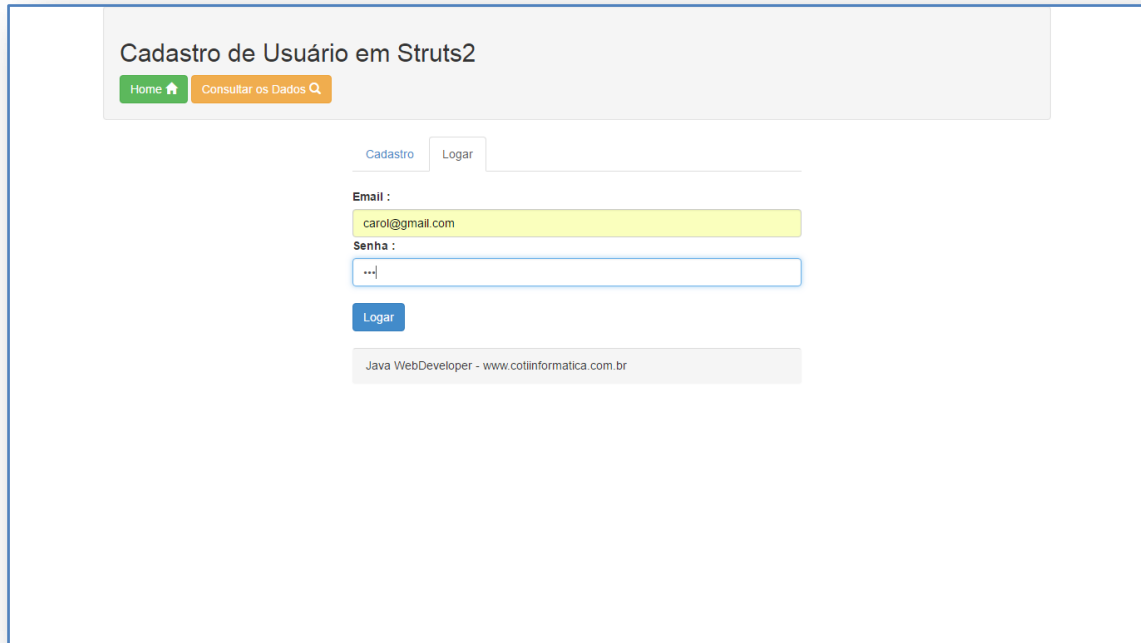
[Home](#) [Consultar os Dados](#)

Email :

Senha :

Java WebDeveloper - www.cotiinformatica.com.br

Logando...



Cadastro de Usuário em Struts2

Home [Consultar os Dados](#)

Cadastro Logar

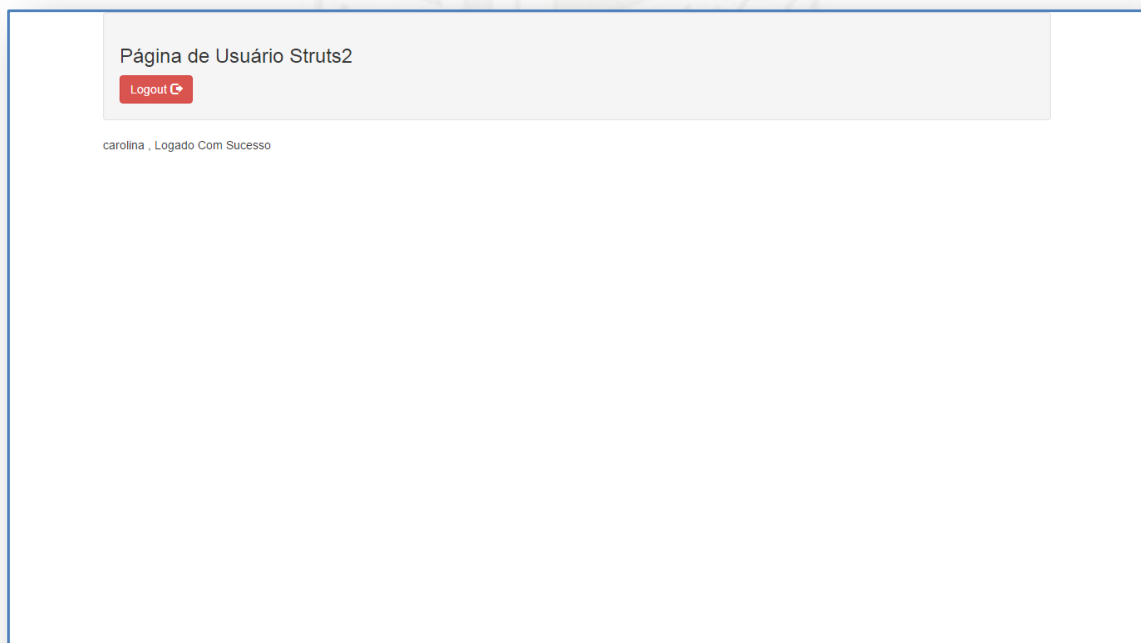
Email :
carol@gmail.com

Senha :

Logar

Java WebDeveloper - www.cotiinformatica.com.br

Logado...



Página de Usuário Struts2

Logout

carolina , Logado Com Sucesso

error.jsp

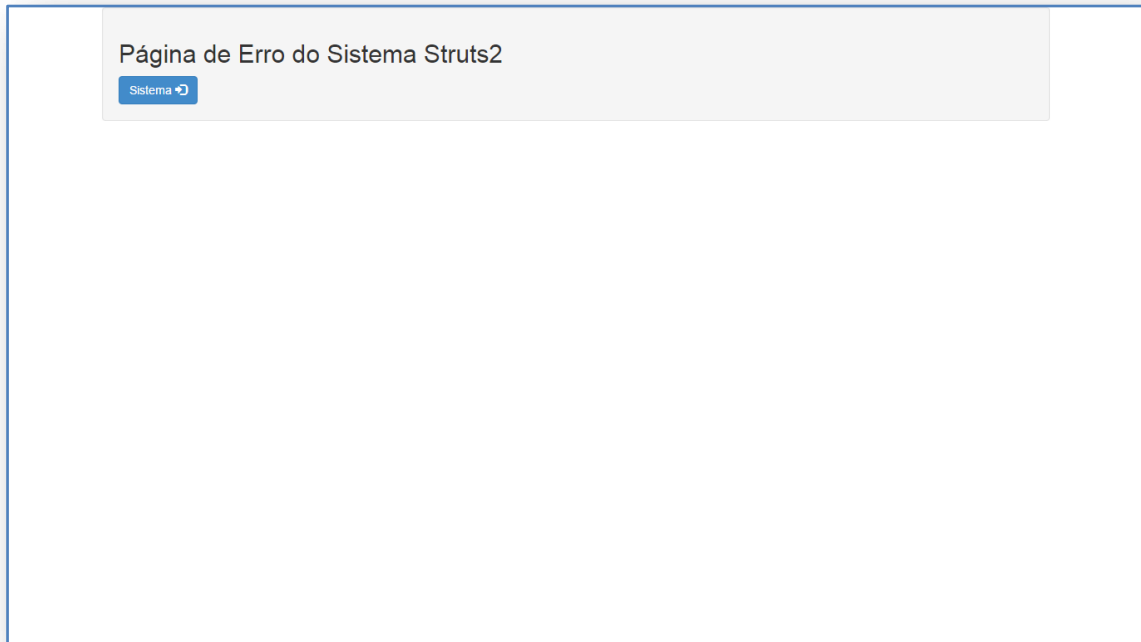
```
<!doctype html>
<html>

<head>
<link rel="stylesheet" href="css/bootstrap.css">
<script type="text/javascript" src="js/jquery-
1.10.2.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
</head>

<body>
  <div class="container">
    <div class="well">
      <h2>Página de Erro do Sistema Struts2</h2>
      <a href="sistema.jsp" class="btn btn-primary">Sistema
      <span class="glyphicon glyphicon-log-in"></span></a>
    </div><!-- FECHA WELL -->

    <div class="col-sm-8 col-sm-offset-2">
      <div>${msg}</div>
    </div><!-- FECHA COL -->
  </div><!-- FECHA CONTAINER -->

</body>
</html>
```



consulta.jsp

```
<!DOCTYPE html>
<html>

<head>
<%@ taglib prefix="s" uri="/struts-tags"%>
<link rel="stylesheet" href="css/bootstrap.css">
<script type="text/javascript" src="js/jquery-
1.10.2.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
</head>

<body>
  <div class="container">
    <div class="well">
      <h2>Consulta Struts Usuários</h2>
      <s:a href="usuario.home.action" cssClass="btn btn-success">Home
      <span class="glyphicon glyphicon-home"></span></s:a>
    </div><!-- FECHA WELL -->

    <div class="col-sm-8 col-sm-offset-2">
      <div class="panel panel-info">
```



```
<div class="panel-heading">
<div class="panel-title">Consulta</div>
</div><!-- FECHA PANEL HEADING -->

<div class="panel-body">
  <table class="table table-hover">
    <tr>
      <td>Nome</td>
      <td>Email</td>
      <td>Perfil</td>
    </tr>
    <s:iterator value="lista">
      <tr>
        <td><s:property value="nome" /></td>
        <td><s:property value="email" /></td>
        <td><s:property value="perfil" /></td>
      </tr>
    </s:iterator>
  </table>

</div><!-- FECHA PANEL-BODY -->
</div><!-- FECHA PANEL-INFO -->

<div class="panel panel-footer">
  <footer>Java WebDeveloper -
www.cotiinformatica.com.br</footer>
</div><!-- FECHA PANEL FOOTER -->
</div><!-- FECHA COL -->
</div><!-- FECHA CONTAINER -->

</body>
</html>
```

Consulta Struts Usuários

[Home](#)

| Nome | Email | Perfil |
|----------|-------------------|--------|
| luciana | luciana@gmail.com | usu |
| carolina | carol@gmail.com | usu |

Java WebDeveloper - www.cotiinformatica.com.br

```
mysql> use struts2;
Database changed
mysql> show tables;
+-----+
| Tables_in_struts2 |
+-----+
| usuario            |
+-----+
1 row in set (0.00 sec)

mysql> desc usuario;
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| idUsuario | int(11) | NO | PRI | NULL | auto_increment |
| email | varchar(50) | NO | UNI | NULL | |
| nome | varchar(50) | NO | | NULL | |
| perfil | varchar(10) | NO | | NULL | |
| senha | varchar(100) | NO | | NULL | |
+----+-----+-----+-----+-----+-----+
5 rows in set (0.03 sec)

mysql> select*from usuario;
+----+-----+-----+-----+-----+
| idUsuario | email | nome | perfil | senha |
+----+-----+-----+-----+-----+
| 1 | luciana@gmail.com | luciana | usu | fc89a7011651885a20cf3851629d8c5f |
+----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select*from usuario;
+----+-----+-----+-----+-----+
| idUsuario | email | nome | perfil | senha |
+----+-----+-----+-----+-----+
| 1 | luciana@gmail.com | luciana | usu | fc89a7011651885a20cf3851629d8c5f |
| 2 | carol@gmail.com | carolina | usu | fc89a7011651885a20cf3851629d8c5f |
+----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> _
```

usu/logado.jsp

```
<!DOCTYPE html>
<html>

<head>
<meta http-equiv="Pragma" content="no-cache">
<meta http-equiv="Cache-Control" content="no-cache">
<meta http-equiv="Expires" content="0">
<meta http-equiv="Cache-Control" content="must-revalidate">

<%@ page import="entity.Usuario"%>
<%@ taglib prefix="s" uri="/struts-tags"%>

<link rel="stylesheet" href="css/bootstrap.css">
<script type="text/javascript" src="js/jquery-1.10.2.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
</head>

<%
    response.setHeader("Cache-Control", "no-cache");
    response.setHeader("Cache-Control", "no-store");
    response.setDateHeader("Expires", -1);
    response.setHeader("Pragma", "no-cache");
%>

<body>
    <div class="container">
        <div class="well">
            <h3>Página de Usuário Struts2</h3>
            <s:a href="usuario.logout.action" cssClass="btn btn-danger">Logout <span class="glyphicon glyphicon-logout"></span> </s:a>
        </div>
        <!-- FECHA WELL -->

        <%
            try {
                if (session.getAttribute("usuario") != null) {
                    Usuario resp = (Usuario) session.getAttribute("usuario");
                    //Imprime o Nome, ScriptLet ...
                    out.print(resp.getNome());
                }
            }
        %>
    </div>
</body>
</html>
```

```
        } catch (Exception ex) {  
            response.sendRedirect("../index.jsp");  
        }  
    %>  
    , ${msg}  
  
</div>  
<!-- FECHA CONTAINER -->  
</body>  
</html>
```

