

Laboratorio Integrador. Página de evaluación de servicios

Objetivo.

Desarrollar una aplicación que permita evaluar los diferentes servicios en una Universidad utilizando diferentes métodos de selección y registrarlos en una base de datos.

Pasos a seguir:

1. Importar una base de datos con una tabla para usuarios y una para las evaluaciones.
2. Desarrollar una vista y clases necesarias para realizar una validación de datos del usuario.
3. Desarrollar una vista y clases necesarias que permitan calificar los servicios de la Universidad.

1. Importar información de una base de datos

- Desarrollar una vista y clases necesarias para la validación de datos del usuario.
- Crear y configurar un nuevo workspace
- Crear los paquetes:
 - **ts.struts.beans**
 - **ts.struts.modelo**
 - **ts.struts.servlets**

Es necesario una javabean en el paquete **ts.struts.beans** que permita leer los datos del usuario de la pantalla.

ValidarUsuarioForm.java

```
package ts.struts.beans;
import org.apache.struts.action.ActionForm;
public class ValidarUsuarioForm extends ActionForm{

    private String login;
    private String passwd;
    public String getLogin() {
        return login;
    }
    public String getPasswd() {
        return passwd;
    }
    public void setLogin(String login) {
        this.login = login;
    }
    public void setPasswd(String passwd) {
        this.passwd = passwd;
    }
}
```

Registrar el javabean en el archivo **struts-config.xml**

```
<form-bean name="OpcionesEncuestaForm" type="ts.struts.beans.OpcionesEncuestaForm"></form-bean>
```

Desarrollo de la vista

login.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Pantalla de validacion de usuario</title>
</head>
<body>
<%session.removeAttribute("ValidarUsuarioForm"); %>
<center>
<h1> Introduce tus datos </h1>

<html:form action="/validar" method="POST">
Login
<html:text property="login"></html:text><br><br>
Password
<html:password property="passwd"></html:password>
<br><br>
<html:submit>Aceptar</html:submit>
<html:reset>Limpiar</html:reset>
</html:form>
</center>
</body>
</html>
```

Desarrollar el ActionServlet que validará al usuario.

ValidarUsuarioAction.java

```
package ts.struts.servlets;

import javax.servlet.http.*;
import org.apache.struts.action.*;
import ts.struts.beans.ValidarUsuarioForm;
import ts.struts.modelo.ValidarUsuario;

public class ValidarUsuarioAction extends Action{
```

```

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

        ValidarUsuarioForm vf = new ValidarUsuarioForm();
        ValidarUsuario valida = new ValidarUsuario();

        String
driver=this.getServlet().getServletContext().getInitParameter("driver");
        String
informacion=this.getServlet().getServletContext().getInitParameter("cadenaCon");
        vf=(ValidarUsuarioForm)form;

        if(valida.validarUsusario(vf, driver, informacion))
            return mapping.findForward("encuesta");

        else
            return mapping.findForward("error");
    }
}

```

Es necesario desarrollar una vista que indique cuando la validación sea incorrecta

error.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Validacion incorrecta</title>
</head>
<body>
<h1>Tus datos son incorrectos</h1>
<html:link forward="login">Volver a la pantalla de validacion</html:link>
</body>
</html>

```

Es necesario registrar el ActionServlet en el archivo **struts-config.xml**

```

<action      path="/validar"      name="ValidarUsuarioForm"      scope="session"
type="ts.struts.servlets.ValidarUsuarioAction">
<forward name="encuesta" path="/paginas/encuesta.jsp"></forward>
<forward name="error" path="/paginas/error.jsp"></forward>
</action>

```

El javabean utilizado para leer los datos de la encuesta se encuentra en el paquete **ts.struts.javabeans**

OpcionesEncuestaForm.java

```
package ts.struts.beans;

import org.apache.struts.action.ActionForm;

public class OpcionesEncuestaForm extends ActionForm{

    private String tipoUsuario;
    private String cafeteria;
    private String biblioteca;
    private String centroComputo;
    private String sistemasEscolares;

    private String [] otrasOpciones;

    private boolean anonimo;

    private String comentario;

    public String getTipoUsuario() {
        return tipoUsuario;
    }

    public String getCafeteria() {
        return cafeteria;
    }

    public String getBiblioteca() {
        return biblioteca;
    }

    public String getCentroComputo() {
        return centroComputo;
    }

    public String getSistemasEscolares() {
        return sistemasEscolares;
    }

    public String[] getOtrasOpciones() {
        return otrasOpciones;
    }

    public boolean isAnonimo() {
        return anonimo;
    }

    public String getComentario() {
```

```

        return comentario;
    }

    public void setTipoUsuario(String tipoUsuario) {
        this.tipoUsuario = tipoUsuario;
    }

    public void setCafeteria(String cafeteria) {
        this.cafeteria = cafeteria;
    }

    public void setBiblioteca(String biblioteca) {
        this.biblioteca = biblioteca;
    }

    public void setCentroComputo(String centroComputo) {
        this.centroComputo = centroComputo;
    }

    public void setSistemasEscolares(String sistemasEscolares) {
        this.sistemasEscolares = sistemasEscolares;
    }

    public void setOtrasOpciones(String[] otrasOpciones) {
        this.otrasOpciones = otrasOpciones;
    }

    public void setAnonimo(boolean anonimo) {
        this.anonimo = anonimo;
    }

    public void setComentario(String comentario) {
        this.comentario = comentario;
    }
}

```

Desarrollar la vista y las clases necesarias para registrar y almacenar las opiniones sobre los servicios de la Universidad

encuesta.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<%@ taglib uri="http://struts.apache.org/tags-bean" prefix="bean" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Pagina de encuesta</title>
</head>
<body>
<html:form action="/registrarRespuestas" method="POST">

```

```

Bienvenido : <bean:write name="ValidarUsuarioForm" property="login"/> <br>

<h1>Elige el tipo de usuario que eres:</h1><br>

<html:select property="tipoUsuario">
<html:option value="profesor">Profesor</html:option>
<html:option value="alumno">Alumno</html:option>
<html:option value="alumnoMaestria">Alumno de la Maestria</html:option>
<html:option value="trabajador">Trabajador</html:option>
</html:select>

<h1>Por favor califica a los siguientes servicios:</h1><br>
<h2>Cafeteria</h2>
MUY BIEN <html:radio property="cafeteria" value="MB"></html:radio>
BIEN<html:radio property="cafeteria" value="B"></html:radio>
SUFICIENTE<html:radio property="cafeteria" value="S"></html:radio>
NO APROBADO<html:radio property="cafeteria" value="NA"></html:radio>
<br>
<h2>Biblioteca</h2>
MUY BIEN <html:radio property="biblioteca" value="MB"></html:radio>
BIEN<html:radio property="biblioteca" value="B"></html:radio>
SUFICIENTE<html:radio property="biblioteca" value="S"></html:radio>
NO APROBADO<html:radio property="biblioteca" value="NA"></html:radio>
<br>
<h2>Centro de computo</h2>
MUY BIEN <html:radio property="centroComputo" value="MB"></html:radio>
BIEN<html:radio property="centroComputo" value="B"></html:radio>
SUFICIENTE<html:radio property="centroComputo" value="S"></html:radio>
NO APROBADO<html:radio property="centroComputo" value="NA"></html:radio>
<br>
<h2>Sistemas escolares</h2>
MUY BIEN <html:radio property="sistemasEscolares" value="MB"></html:radio>
BIEN<html:radio property="sistemasEscolares" value="B"></html:radio>
SUFICIENTE<html:radio property="sistemasEscolares" value="S"></html:radio>
NO APROBADO<html:radio property="sistemasEscolares" value="NA"></html:radio>

<h1>Crees que la Universidad necesita mas: </h1><br>

<html:multibox property="otrasOpciones" value="verdes"></html:multibox>Areas
verdes
<html:multibox property="otrasOpciones" value="mesas"></html:multibox>Mesas en
las explanadas
<html:multibox property="otrasOpciones"
value="deportes"></html:multibox>Actividades deportivas
<html:multibox property="otrasOpciones"
value="culturales"></html:multibox>Actividades culturales

<h1>Escribe un comentario</h1><br>

<html:textarea property="comentario" rows="10" cols="30"></html:textarea><br>
<html:checkbox property="anonimo">Comentario anonimo </html:checkbox><br><br>

<html:submit>Registrar</html:submit>

```

```

</html:form>
</body>
<html:link forward="login">Salir del sistema</html:link>
</html>

```

El ActionServlet que registra las opciones elegidas en la encuesta se creará en el paquete **ts.struts.servlets**

RegistrarRespuestasAction.java

```

package ts.struts.servlets;

import javax.servlet.http.*;
import org.apache.struts.action.*;

import ts.struts.beans.*;
import ts.struts.modelo.*;

public class RegistrarRespuestasAction extends Action{

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

        HttpSession sesion = request.getSession();

        OpcionesEncuestaForm ef = new OpcionesEncuestaForm();
        ValidarUsuarioForm uf =
(ValidarUsuarioForm)sesion.getAttribute("ValidarUsuarioForm");

        ef=(OpcionesEncuestaForm) form;

        String
driver=this.getServlet().getServletContext().getInitParameter("driver");
        String
informacion=this.getServlet().getServletContext().getInitParameter("cadenaCon");

        RegistrarRespuestasEncuesta registra = new
RegistrarRespuestasEncuesta();
        registra.registrarOpciones(ef, uf, driver, informacion);

        return mapping.findForward("gracias");
    }
}

```

También es necesario desarrollar una vista que despliegue que el registro fue exitoso

agradecimiento.jsp

```

<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>

```

```

<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Pantalla de agradecimiento</title>
</head>
<body>
<h1>Gracias por tu opiniÃ³n!!, Ã©sta serÃ¡ registrada pero no tomada en
cuenta</h1>
</body>
<html:link forward="login">Salir del sistema</html:link>
</html>

```

El ActionServlet debe ser registrado en el archivo **struts-config.xml**

```

<action path="/registrarRespuestas" name="OpcionesEncuestaForm" scope="request"
type="ts.struts.servlets.RegistrarRespuestasAction">
<forward name="gracias" path="/paginas/agradecimiento.jsp"></forward>
</action>

```

Finalmente el archivo **struts-config.xml** queda de la siguiente manera:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE struts-config PUBLIC "-//Apache Software Foundation//DTD Struts
Configuration 1.1//EN" "http://jakarta.apache.org/struts/dtds/struts-
config_1_1.dtd">
<struts-config>
  <data-sources />
  <form-beans>
    <form-bean name="OpcionesEncuestaForm"
type="ts.struts.beans.OpcionesEncuestaForm"></form-bean>
    <form-bean name="ValidarUsuarioForm"
type="ts.struts.beans.ValidarUsuarioForm"></form-bean>
  </form-beans>
  <global-exceptions />
  <global-forwards>
    <forward name="login" path="/paginas/login.jsp"></forward>
  </global-forwards>
  <action-mappings>
    <action path="/validar" name="ValidarUsuarioForm" scope="session"
type="ts.struts.servlets.ValidarUsuarioAction">
      <forward name="encuesta" path="/paginas/encuesta.jsp"></forward>
      <forward name="error" path="/paginas/error.jsp"></forward>
    </action>
    <action path="/registrarRespuestas" name="OpcionesEncuestaForm"
scope="request" type="ts.struts.servlets.RegistrarRespuestasAction">
      <forward name="gracias" path="/paginas/agradecimiento.jsp"></forward>
    </action>
  </action-mappings>
  <controller bufferSize="4096" debug="0" />
</struts-config>

```



```
<message-resources parameter="com.yourcompany.struts.ApplicationResources" />
</struts-config>
```

Finalmente se deben desarrollar las clases que permitan el manejo de la base de datos, pero antes es necesario registrar la información correspondiente a la base de datos en el archivo **web.xml**

web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app id="WebApp_ID" version="2.4" xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://
java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
  <display-name>
  TiposDeSeleccion</display-name>
  <servlet>
    <servlet-name>action</servlet-name>
    <servlet-class>org.apache.struts.action.ActionServlet</servlet-class>
    <init-param>
      <param-name>config</param-name>
      <param-value>/WEB-INF/struts-config.xml</param-value>
    </init-param>
    <init-param>
      <param-name>debug</param-name>
      <param-value>2</param-value>
    </init-param>
    <init-param>
      <param-name>detail</param-name>
      <param-value>2</param-value>
    </init-param>
    <load-on-startup>2</load-on-startup>
  </servlet>
  <servlet-mapping>
    <servlet-name>action</servlet-name>
    <url-pattern>*.do</url-pattern>
  </servlet-mapping>
  <session-config>
    <session-timeout>30</session-timeout>
  </session-config>
  <welcome-file-list>
    <welcome-file></welcome-file>
  </welcome-file-list>
  <context-param>
    <param-name>driver</param-name>
    <param-value>com.mysql.jdbc.Driver</param-value>
  </context-param>
  <context-param>
    <param-name>cadenaCon</param-name>
    <param-value>jdbc:mysql://localhost/integrador_2</param-value>
  </context-param>
</web-app>
```

Las clases necesarias se crearán en el paquete **ts.struts.modelo**

ManejoDeBaseDeDatos.java

```
package ts.struts.modelo;
import java.sql.*;
public class ManejoDeBaseDeDatos {

    private Connection conexion=null;
    private String driver;
    private String informacion;

    public ManejoDeBaseDeDatos(String d, String info) {

        this.driver=d;
        this.informacion=info;
    }
    public Connection abrirConexion(){

        try{
            Class.forName(driver).newInstance();
            conexion=DriverManager.getConnection(informacion, "root",
"root");
        }catch(Exception e){
            e.printStackTrace();
        }

        return conexion;
    }
    public void cerrarConexion() {
        try {
            conexion.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

ValidarUsuario.java

```
package ts.struts.modelo;

import ts.struts.beans.*;
import java.sql.*;

public class ValidarUsuario {

    public boolean validarUsusario(ValidarUsuarioForm vf, String d, String i){

        ManejoDeBaseDeDatos base = new ManejoDeBaseDeDatos(d,i);
        ValidarUsuarioForm registro = new ValidarUsuarioForm();
    }
}
```

```

        Connection con = base.abrirConexion();
        Statement s;
        boolean encontrado;

        try{
            s=con.createStatement();
            ResultSet rs = s.executeQuery("SELECT * FROM usuarios WHERE
login='" +vf.getLogin() +"'");

            if(!rs.first()){
                encontrado=false;;
            }

            else{
                String pb = rs.getString("passwd");

                if(pb.compareTo(vf.getPasswd())==0)
                    encontrado=true;
                else
                    encontrado=false;
            }
            s.close();
            base.cerrarConexion();
            return encontrado;
        }catch (Exception e) {
            e.printStackTrace();
            return false;
        }

    }
}

```

Para el registro, se colocan valores por defecto en caso de que no se elija ninguno

RegistrarRespuestasEncuesta.java

```

package ts.struts.modelo;

import java.sql.Connection;
import java.sql.Statement;

import ts.struts.beans.OpcionesEncuestaForm;
import ts.struts.beans.ValidarUsuarioForm;

public class RegistrarRespuestasEncuesta {

    public void registrarOpciones(OpcionesEncuestaForm ef,
                                   ValidarUsuarioForm uf,
                                   String drv, String inf){
    }
}

```

```

ManejoDeBaseDeDatos base = new ManejoDeBaseDeDatos(drv,inf);
Connection con = base.abrirConexion();
Statement s;

String verdes="No";
String mesas="No";
String deportes="No";
String cultura="No";
String cafeteria;
String biblioteca;
String centroComputo;
String sistemasEscolares;
String comentario;
String tipoUsuario;
String login;

if(ef.getCafeteria()==null)
    cafeteria="MB";
else
    cafeteria=ef.getCafeteria();

if(ef.getBiblioteca()==null)
    biblioteca="MB";
else
    biblioteca=ef.getBiblioteca();

if(ef.getCentroComputo()==null)
    centroComputo="MB";
else
    centroComputo=ef.getCentroComputo();

if(ef.getSistemasEscolares()==null)
    sistemasEscolares="MB";
else
    sistemasEscolares=ef.getSistemasEscolares();
if(ef.getComentario()==null)
    comentario="";
else
    comentario=ef.getComentario();
if(ef.isAnonimo())
    login="anonimo";
else
    login=uf.getLogin();

if(ef.getOtrasOpciones()!=null){
for (int i=0;i<ef.getOtrasOpciones().length;i++){
    if(ef.getOtrasOpciones()[i].compareTo("verdes")==0)
        verdes="Si";
    if(ef.getOtrasOpciones()[i].compareTo("mesas")==0)
        mesas="Si";
    if(ef.getOtrasOpciones()[i].compareTo("deportes")==0)
        deportes="Si";
    if(ef.getOtrasOpciones()[i].compareTo("culturales")==0)
        cultura="Si";
}
}

```

```

    }
    }

    tipoUsuario=ef.getTipoUsuario();

    try{
        s=con.createStatement();
        s.executeUpdate("INSERT INTO evaluacion" +
            "(cafeteria,biblioteca,computo,sistemas,"+
            "areasverdes,mesasplaza,deportes,culturales," +
            "comentario,tipousuario,login)" +
            " VALUES (" +
            "'" +cafeteria+"'," +
            "'" +biblioteca+"'," +
            "'" +centroComputo+"'," +
            "'" +sistemasEscolares+"'," +
            "'" +verdes+"'," +
            "'" +mesas+"'," +
            "'" +deportes+"'," +
            "'" +cultura+"'," +
            "'" +comentario+"'," +
            "'" +tipoUsuario+"'," +
            "'" +login+"')");
    }catch (Exception e) {
        e.printStackTrace();
    }
}

}

```

Probar la aplicación ejecutando el archivo login.jsp en el Servidor