

Bienvenido

Nombre de usuario

Contraseña

Iniciar sesión

[Olvidó su contraseña?](#)

[Nuevo usuario?](#)

[Regístrese aquí](#)

JSF - Internacionalización



Escrito por Elmer Kristopher Zapata Ramírez

viernes, 30 junio 2006

INTRODUCCIÓN

OBJETIVO

REQUISITOS

MANOS A LA OBRA

Paso 1: Creación del Proyecto

Paso 2: Creación de los archivos de Recursos

Paso 3: Creación de las Páginas Web

Paso 4: Creación del Bean (IdiomaBean)

Paso 5: Edición de las Páginas Webs.

Paso 6: Ejecución de la Aplicación

ACERCA DEL AUTOR

Nota: para ver las imágenes con mayor tamaño deben descargarlas a tu PC. Haciendo clic con el botón derecho del mouse y eligiendo la opción "Guardar como ..."

INTRODUCCIÓN

Hoy en día los mercados extendidos y la misma globalización a permitido que clientes no locales puedan acceder a realizar transacciones comerciales a nuestros negocios las 24 horas del día , los 7 días de la semana (24x7) , siendo las horas comerciales locales irrelevantes para la web, es por ello que los desarrolladores han buscado la forma de dar soporte a estos, con independencia geográfica e idiomática de la forma mas simple posible.

OBJETIVO

El objetivo de este artículo es introducirnos al desarrollo de aplicaciones Web que den soporte a múltiples usuarios con independencia geográfica e idiomática, para ello desarrollaremos una pequeña aplicación de Login sin conexión a base de datos usando paginas jsp que nos permita la Internacionalización, para lo cual usaremos el Marco de trabajo Java Server Face en JDeveloper 10G

Encuesta

¿Ya está utilizando JSF o ADF Faces en sus proyectos ?

SI

No, todavía uso JSP puro.

No, todavía uso JSP con Struts.

No, todavía uso JSP, Struts, JSTL y Binding.

Ultimos Artículos

JSF - Lifecycle **Nuevo!**

JSF - Jerarquía Tree **Nuevo!**

JSF - Internacionalización **Nuevo!**

ProyectoWeb, Jdeveloper y JasperReport

JSF - CREANDO UN LOGIN

Componente Tree de UIX

Proyecto Web - Parte II (Business Layer)

Artículos Destacados

Curso de ADF BC, JSP, Struts con Jdeveloper 10.1.2

ProyectoWeb, JDeveloper y JasperReport

JSF - Creando un login Proyecto Web - Parte II (BUSINESS LAYER)

Introducción a Business Component - Parte I

Buscar Artículos

Usuarios Registrados



Registrados: 3898

REQUISITOS

JDeveloper 10g (10.1.3).

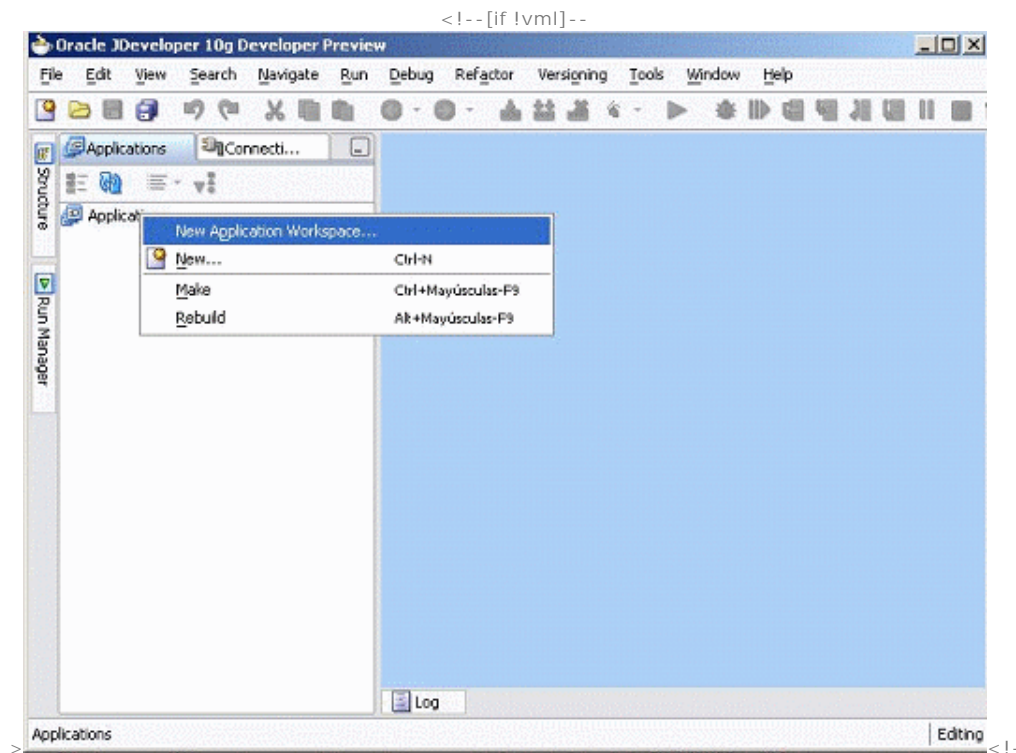
La versión 10.1.3 de JDeveloper la cual pueden descargar de www.oracle.com, nos permite crear gráficamente e intuitivamente aplicaciones Web usando el Framework JAVA SERVER FACE.

MANOS A LA OBRA

Paso 1: Creación del Proyecto

Para crear el proyecto hacemos clic derecho en "Applications" y seleccionamos "New Application Workspace" (ver Figura Nro. 1)

Luego nos aparece una ventana en la cual nos pedirá información de la aplicación que deseamos crear (Ver Figura Nro. 2), Para este caso la aplicación se llamara "InternacionalizacionJSF", luego escogeremos la ruta en la cual se guardara y en "Application Template" seleccionamos "Web Application [JSF, JSP, EJB]."



-[endif]-->

Figura Nro. 1

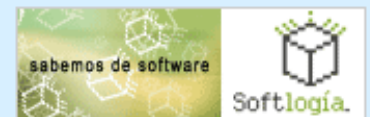
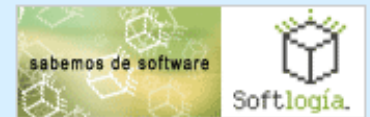
<!--[if !vml]-->

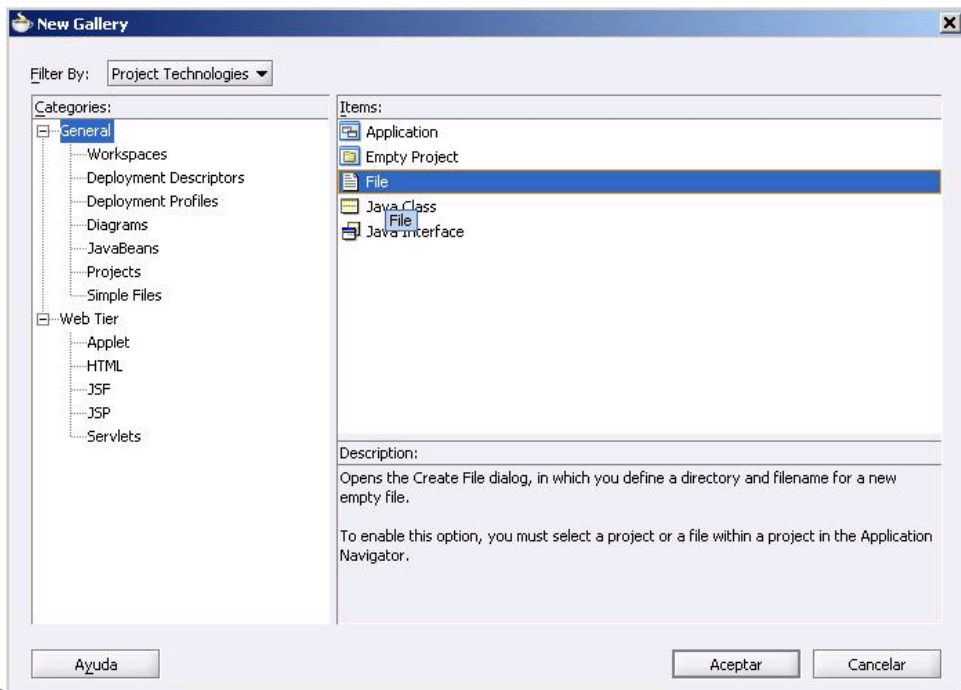
usuarios.

 Este mes: 55 usuarios.

RSS Disponible

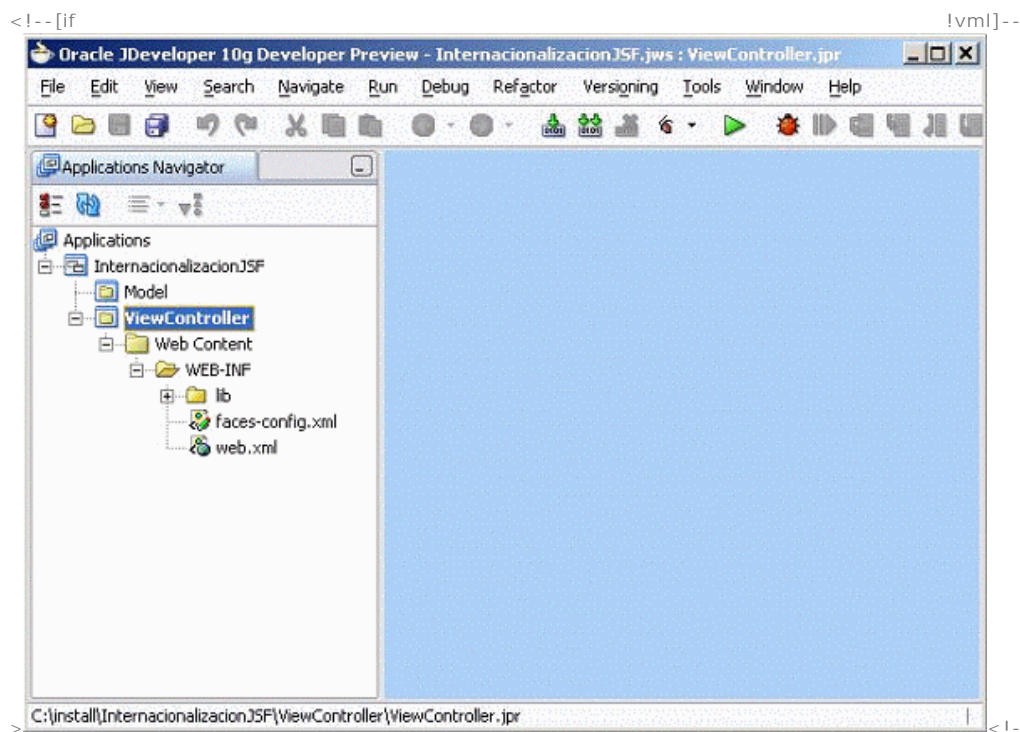
RSS	0.91
RSS	1.0
RSS	2.0
ATOM	0.3





-[endif]-->
Figura Nro. 2

Una vez creada la nueva aplicación quedara como muestra la Figura Nro. 3



-[endif]-->
Figura Nro. 3

Siendo esta la estructura básica de toda aplicación usando esta tecnología JSF, donde el archivo faces-config.xml es el archivo de configuración y en la carpeta lib habrá 2 archivos jar que serán los que contengan las clases usadas por JSF.

Paso 2: Creación de los archivos de Recursos

Para hacer posible la internacionalización nos basaremos en los archivos de recursos los cuales contendrán los valores definidos en sus respectivos idiomas, estos archivos tienen una estructura de CLAVE=VALOR, en donde la clave debe ser la misma para todos los archivos y el valor la traducción para el idioma seleccionado.

Primero hacemos clic derecho sobre [Web Content] - New, en la ventana que aparece seleccionamos [General] - Simple Files - y en la parte derecha nos aparece las opciones a

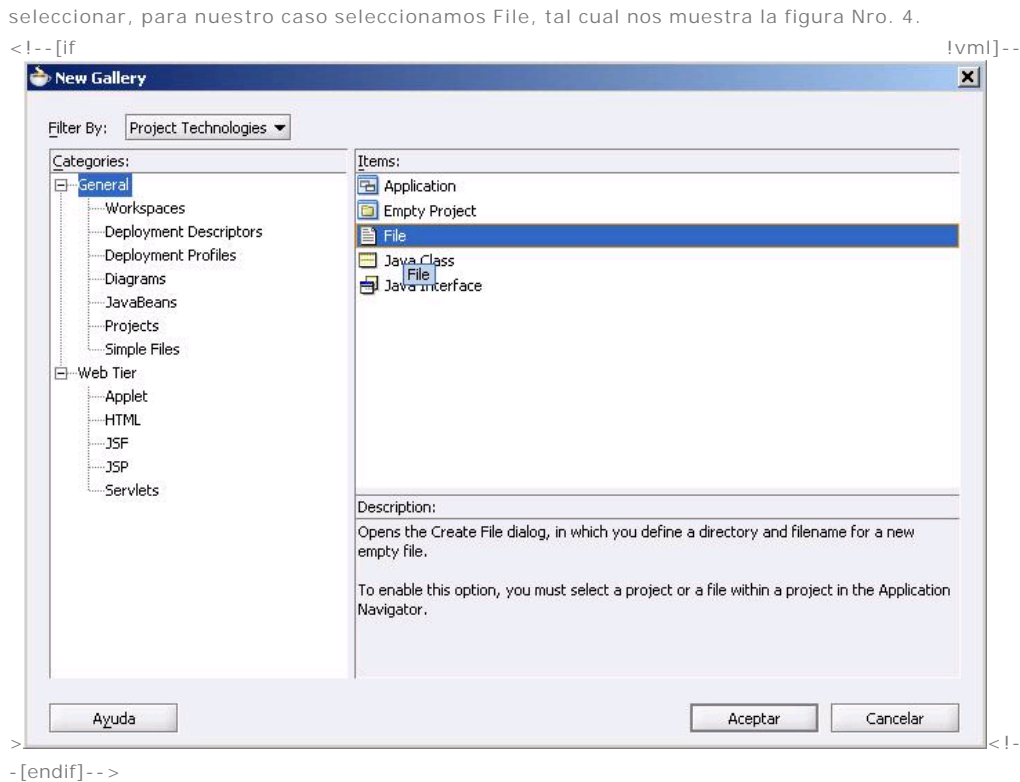


Figura Nro. 4

Los archivos de recursos deben tener una nomenclatura específica para que sea ubicado rápidamente, para ello en su nomenclatura se inicia con un nombre para el archivo de recursos seguido de los códigos del idioma y del país, para más información respecto a códigos de idiomas ir a <http://www.unicode.org/unicode/onlinedat/languages.html> y códigos de países ir a <http://www.unicode.org/unicode/onlinedat/countries.html>

Para nuestros ejemplos creamos 3 archivos (Ver Figura Nro. 5), siendo el primero el archivo por defecto a ser cargado por la aplicación en caso no se seleccione algún otro, recordar que todos deben tener la extensión *.properties.

Message → Español (Idioma por defecto)
 Message_de_De → Alemán
 Message_en_US → Inglés

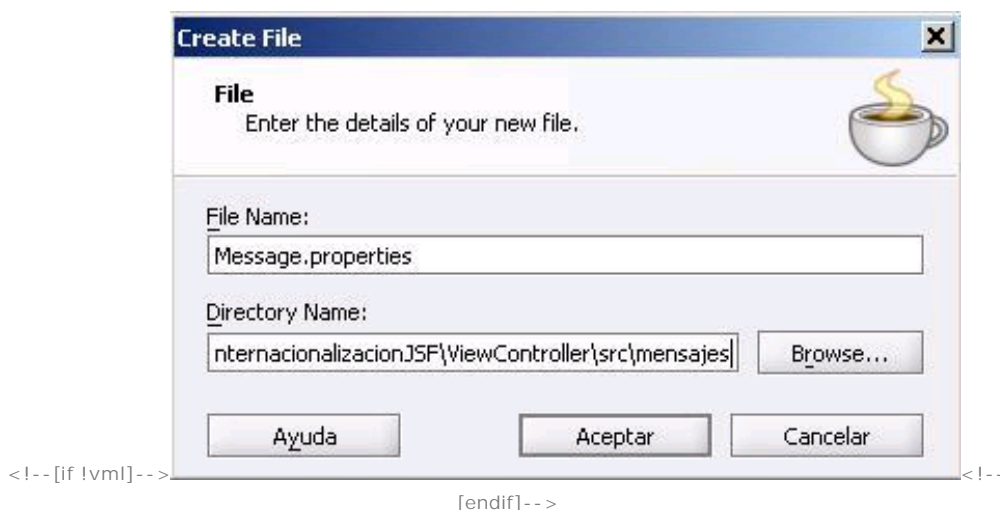


Figura Nro. 5

Después de crear los archivos nuestra aplicación quedará como muestra la Figura Nro. 6

```
<!-- [if !vml]-->
```

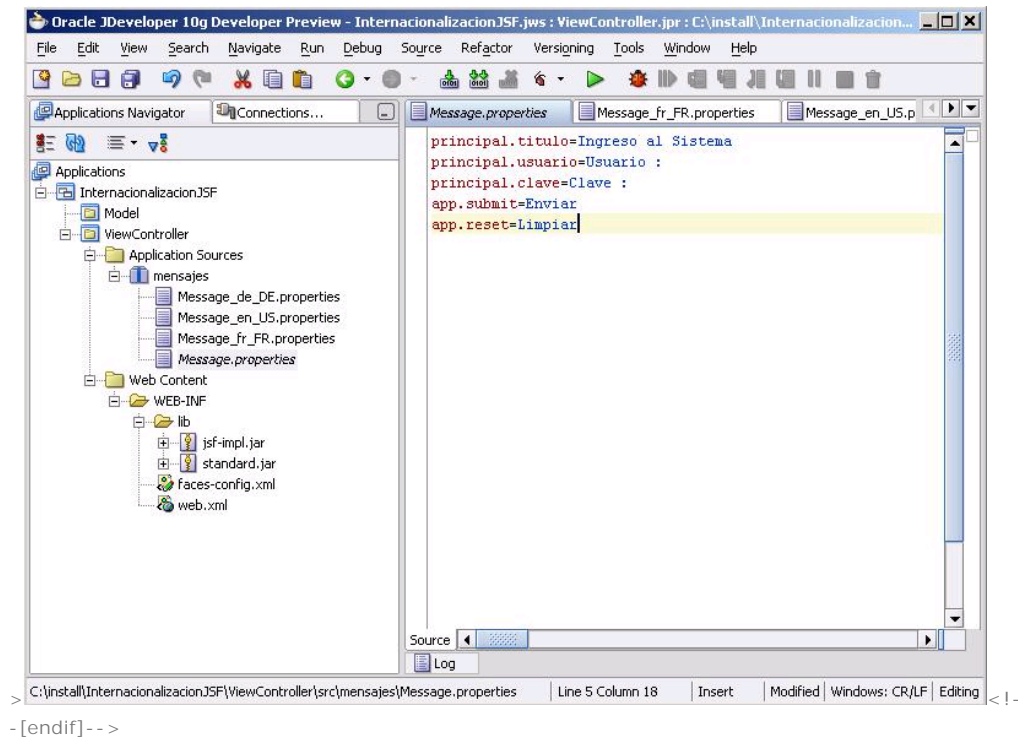


Figura Nro. 6

Paso 3: Creación de las Páginas Web

Para este ejemplo crearemos una página Web, index.jspx, la cual será usada en la aplicación como pagina de acceso al sistema, la idea es tener un Login, pero con la capacidad de personalizar el idioma en el cual se muestre la misma.

Primero hacemos clic derecho sobre [Web Content] - New, en la ventana que aparece seleccionamos [Web Tier] – JSP y en la parte derecha nos aparece las opciones de JSP que se pueden crear, para nuestro caso seleccionamos JSP (pagina Java Server Page), tal cual nos muestra la figura Nro. 7.

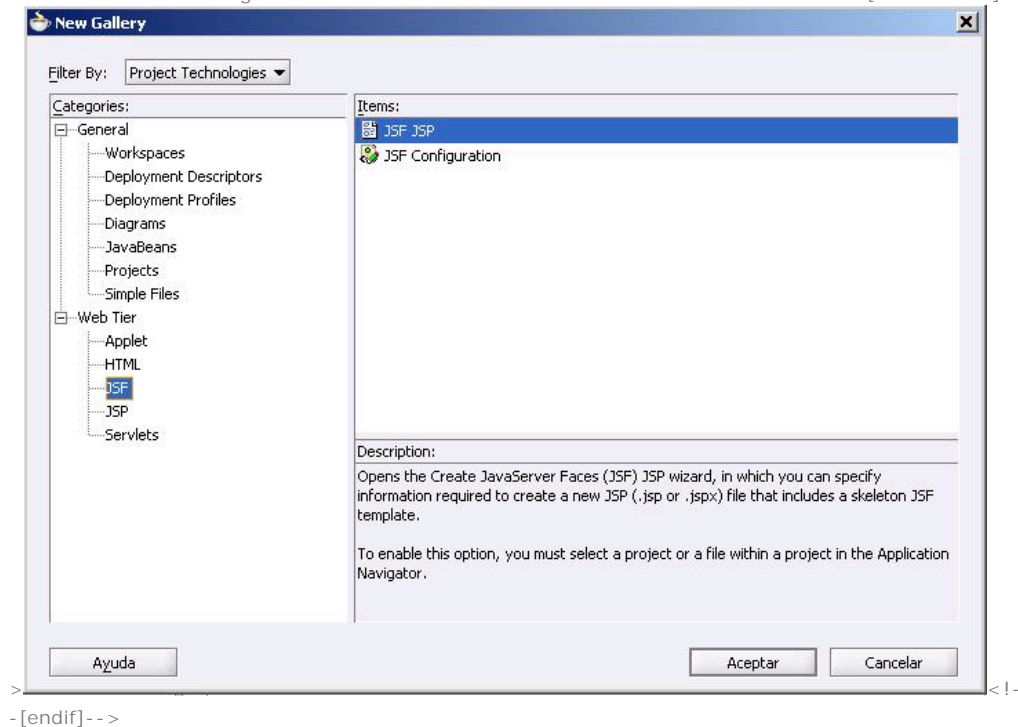
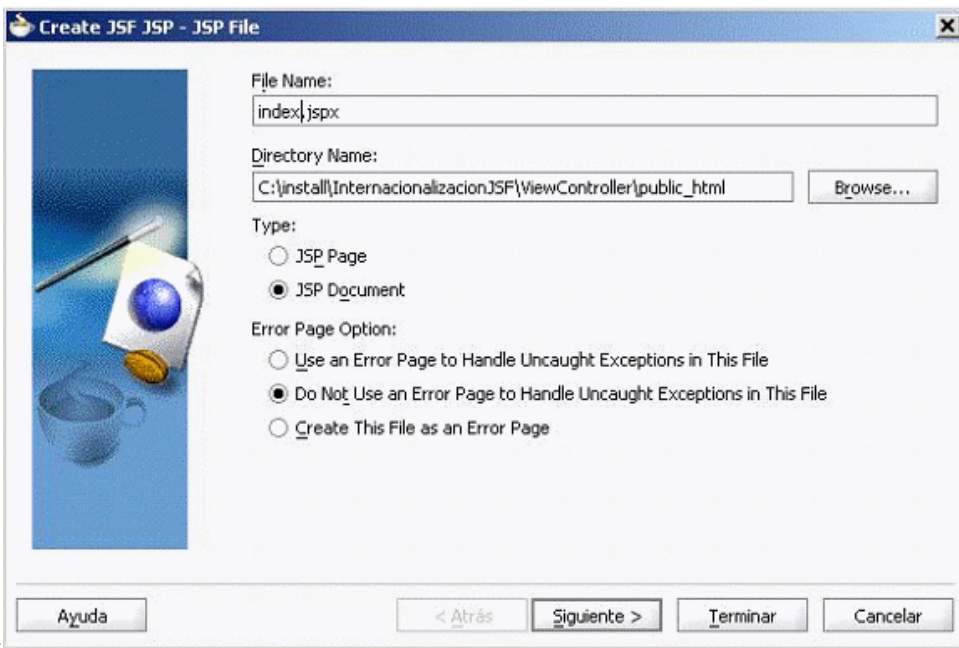


Figura Nro. 7

Al hacer clic en Aceptar nos muestra una ventana similar a la Figura Nro 8 en la cual debemos proporcionar el nombre del archivo (File Name) a crear (index.jspx) , el tipo (Type), para nuestro caso crearemos JSP Document (jspx) y por ultimo en la sección Error Page Option seleccionamos "Do Not Use a Error Page to Handle Uncaught Exception in this File".



- [endif] -->
 Figura Nro. 8

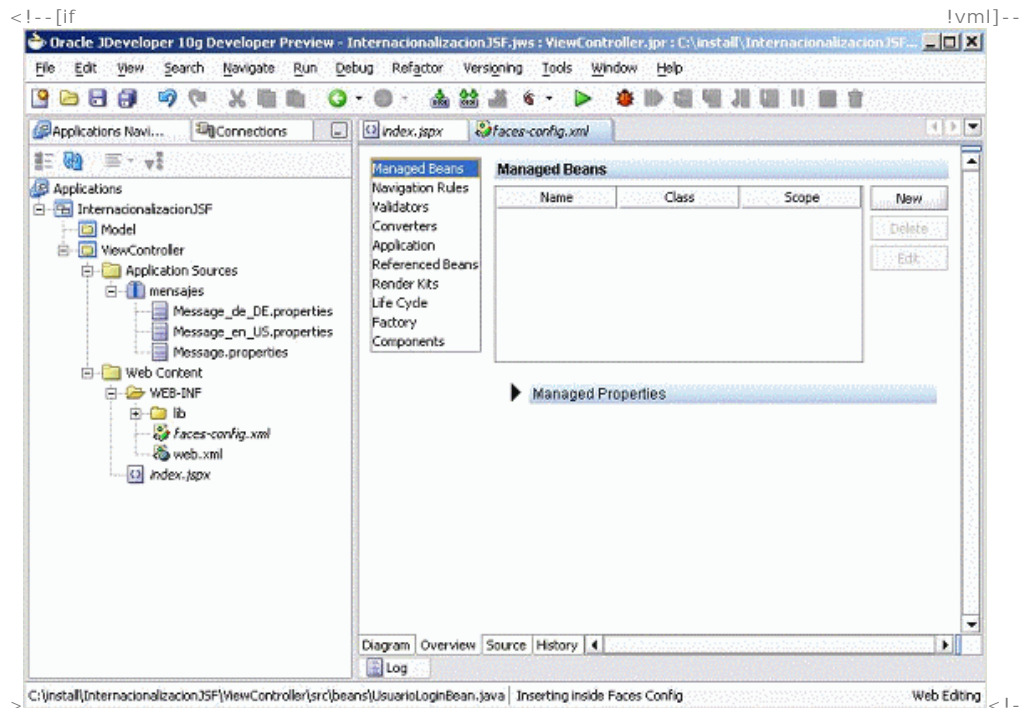
Luego le damos en terminar y aparece la pagina index.jspx dentro del proyecto creado.

Paso 4: Creación del Bean (IdiomaBean)

JDeveloper 10G además de permitirnos diagramar la navegabilidad de nuestras páginas, nos permite editar el archivo faces-config.xml y desde hay crear beans si fuese necesario, como lo será para nuestro ejemplo.

Al hacer doble clic en el archivo faces-config.xml vemos como en el diagramador en la parte inferior existen 4 paletas (Diagram, Overview, Source y History), todas nos permiten ver dicho archivo de distinta manera, ayudando en gran medida a la configuración del archivo.

Para nuestro caso, al hacer Clic en la paleta Overview (Ver Figura Nro. 9) nos muestra varias opciones, siendo una de ellas el Managed Beans, al pulsar sobre la opción Managed Beans, nos muestra en la parte derecha las sub opciones New, Delete y Edit.

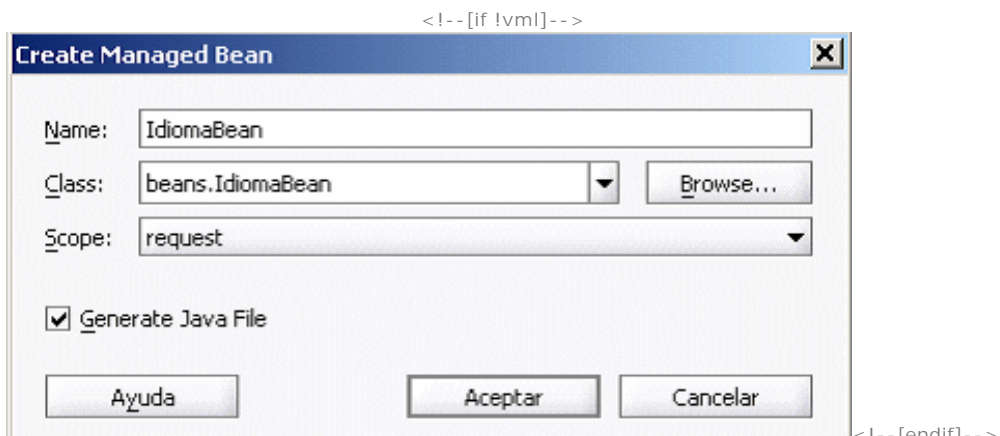


(Figura Nro. 9)

Al seleccionar New nos muestra una ventana desde la cual crearemos un Bean, al cual

llamaremos IdiomaBean y estará en el paquete llamado beans, tal cual lo muestra la Figura Nro. 10.

Obs. Debemos cerciorarnos que antes de pulsar Aceptar debemos haber seleccionado la opción "Generate Java File", la cual nos creara físicamente el archivo IdiomaBean.java dentro de la carpeta beans.



(Figura Nro. 10)

Una vez creado el bean IdiomaBean debemos editarlo y escribir lo siguiente:

```
<!--[if !vml]-->
package beans;

import java.util.Locale;
import javax.faces.context.FacesContext;

public class IdiomaBean
{
    public String alemanAction() {
        FacesContext context = FacesContext.getCurrentInstance();
        context.getViewRoot().setLocale(Locale.GERMAN);
        return null;
    }

    public String inglesAction() {
        FacesContext context = FacesContext.getCurrentInstance();
        context.getViewRoot().setLocale(Locale.ENGLISH);
        return null;
    }
}
<!--[endif]-->
```

Ahora al ir a la paleta Source podemos ver el contenido del archivo faces-config.xml el cual deberá ser como este:

```
<!--[if !vml]-->
```

```

<?xml version="1.0" encoding="windows-1252"?>
<!DOCTYPE faces-config PUBLIC
    "-//Sun Microsystems, Inc.//DTD JavaServer Faces Config 1.1//EN"
    "http://java.sun.com/dtd/web-facesconfig_1_1.dtd">
<faces-config xmlns="http://java.sun.com/JSF/Configuration">
    <managed-bean>
        <managed-bean-name>IdiomaBean</managed-bean-name>
        <managed-bean-class>beans.IdiomaBean</managed-bean-class>
        <managed-bean-scope>request</managed-bean-scope>
    </managed-bean>
    <navigation-rule>
        <from-view-id>/index.jsp</from-view-id>
    </navigation-rule>
</faces-config>
<!--
    -[endif]-->

```

Paso 5: Edición de las Páginas Webs.

Para la edición final de la página index.jsp, Jdeveloper 10G también nos facilita el trabajo con un editor de páginas Web teniendo en consideración la tecnología seleccionada, en nuestro caso la página Web la desarrollaremos con los Tag Pre-Definidos de Java Server Face.

Para editar la página index.jsp le damos doble clic y la editamos visualmente hasta obtener el siguiente diseño mostrado en la Figura Nro. 11, como pueden ver en la parte derecha tienen la paleta JSF HTML, la cual tiene los objetos Web necesarios para la creación de interfaces Web.

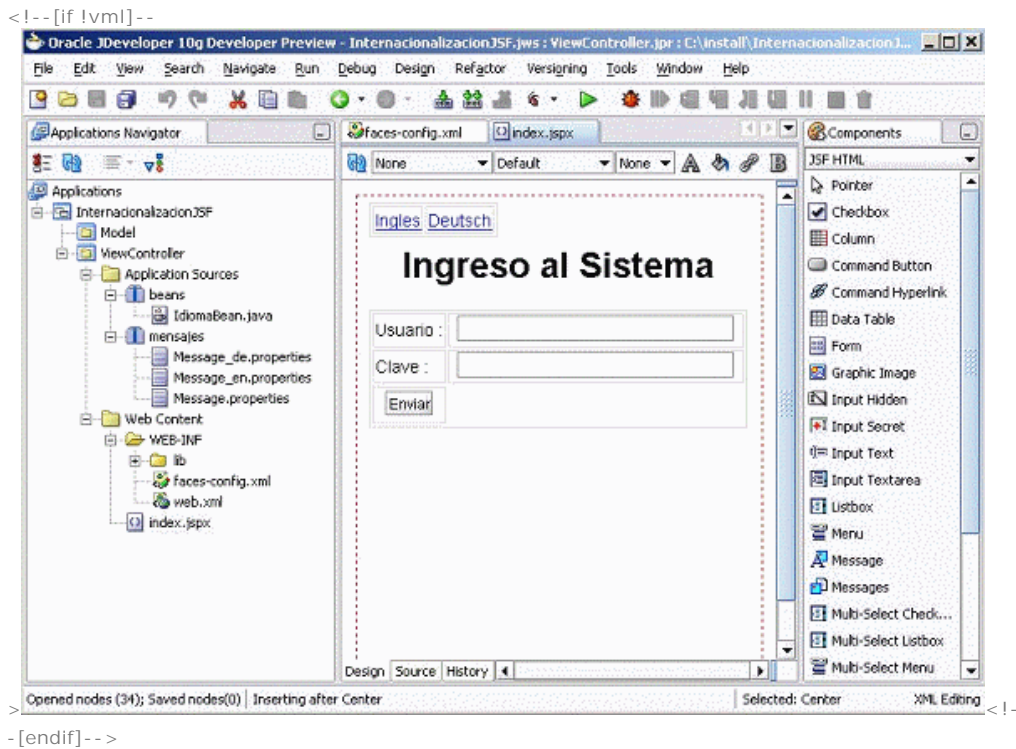


Figura Nro. 11

Cuando pulsamos en Source, podemos apreciar el editor de código de páginas Web que tiene el Jdeveloper 10G el cual nos permite vía código, editar nuestras páginas Web.

Al ver en el editor de código la página index.jsp se muestra el siguiente código:

```

<?xml version='1.0' encoding='windows-1252'?>
<jsp:root xmlns:jsp="http://java.sun.com/JSP/Page" version="2.0"
    xmlns:f="http://java.sun.com/jsf/core"
    xmlns:h="http://java.sun.com/jsf/html">
    <jsp:output omit-xml-declaration="true" doctype-root-element="HTML"
        doctype-system="http://www.w3.org/TR/html4/loose.dtd"
        doctype-public="-//W3C//DTD HTML 4.01 Transitional//EN"/>
    <jsp:directive.page contentType="text/html; charset=windows-1252"/>

```

```

<html>
<f:view>
  <head>
    <f:loadBundle basename="mensajes.Message" var="msgs"/>
    <title>Internacionalización</title>
  </head>
  <body>
    <h:form>
      <table>
        <tr>
          <td>
            <h:commandLink immediate="true"
              action="#{ IdiomaBean.inglesAction}" value="Ingles">
            </h:commandLink>
          </td>
          <td>
            <h:commandLink immediate="true"
              action="#{ IdiomaBean.alemanAction}" value="Deutsch">
            </h:commandLink>
          </td>
        </tr>
      </table>
      <center>
        <h1><h:outputText value="#{msgs.titulo}"/></h1>
      <table>
        <tr>
          <td>
            <h:outputText value="#{msgs.usuario}"/>
          </td>
          <td>
            <h:inputSecret value=""/>
          </td>
        </tr>
        <tr>
          <td>
            <h:outputText value="#{msgs.clave}"/>
          </td>
          <td>
            <h:inputSecret value=""/>
          </td>
        </tr>
        <tr>
          <td>
            <h:commandButton value="#{msgs.submit}"/>
          </td>
        </tr>
      </table>
    </center>
  </h:form>
</body>
</f:view>
</html>
</jsp:root>

```

Donde las primeras 9 líneas y la última son agregadas por Jdeveloper 10G por tratarse de Páginas de tipo JSP Document (*.jspx), dentro de las cuales se hace el llamado a los Tabs de JSF, dándoles un alias de "f" (JSF CORE) y "h" (JSF HTML).

Paso 6: Ejecución de la Aplicación

Como paso final solo queda ejecutar nuestra aplicación para ver los resultados obtenidos, para lo cual seleccionamos la página index.jsx y pulsamos F11 dando como resultado la visualización en el Browser de nuestra página (ver Figura Nro 12)

```
<!--[if
```

```
!vml]--
```

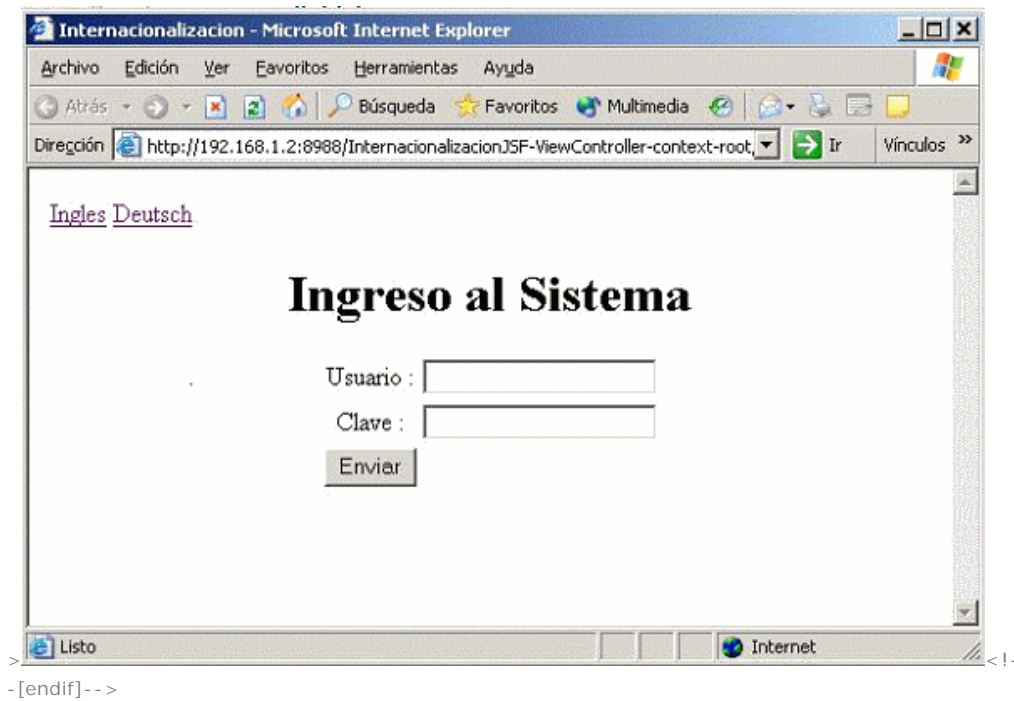


Figura Nro.12

En nuestro caso pulsamos en cualquiera de los links (Ingles o Alemán), al pulsar "Alemán" nos muestra la misma pagina index.jspx pero en idioma Alemán (Ver Figura Nro. 13).

<!--[if !vml]-->

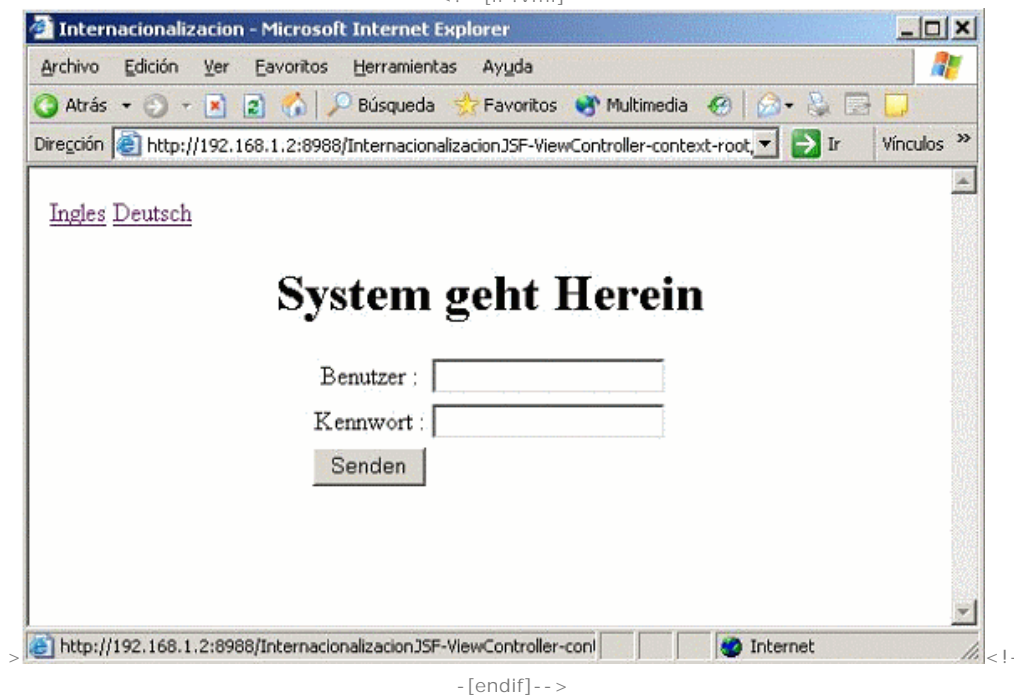


Figura Nro 13

<!--[if !vml]--><!--[endif]-->



Elmer Zapata Ramírez

Es egresado de la Maestría de la Universidad Nacional Mayor de San Marcos - UNMSM en Ingeniería de Sistemas e Informática en la mención de Dirección y Gestión de Tecnologías de Información, es actualmente profesor en tecnologías J2EE, habiendo trabajado en proyectos empresariales con dicha tecnología.

Contactar en: [Esta dirección de correo electrónico está siendo protegida de \"spam bots\", necesitas habilitar Javascript para poder verlo.](#) o en [Esta dirección de correo electrónico está siendo protegida de \"spam bots\", necesitas habilitar Javascript para poder verlo.](#)

Comentario[s]

Sólo los usuarios registrados pueden escribir comentarios.

Por favor identificate o regístrate.

Powered by [AkoComment 2.0!](#)

[< Anterior](#)

[Siguiente >](#)

[\[Atrás \]](#)