



JAVASERVER FACES

NAVEGACIÓN Y LENGUAJE DE EXPRESIONES

Ing. Eric Gustavo Coronel Castillo

gcoronelc@gmail.com

gcoronelc.blogspot.com

AGENDA

- ❖ Navegación
- ❖ Demo
- ❖ Lenguaje de Expresiones
- ❖ Demo



❖ NAVEGACIÓN CLASICA

Desde el inicio de la especificación, JSF 1.x cualquier caso de navegación por trivial que fuese, requería una entrada en el fichero faces-config.xml. Cuando se navegaba de page1.xhtml a page2.xhtml en respuesta a un éxito en un componente, se introducía el siguiente código XML:

```
<navigation-rule>
  <from-view-id>/page1.xhtml</from-view-id>
  <navigation-case>
    <from-outcome>success</from-outcome>
    <to-view-id>/page2.xhtml</to-view-id>
  </navigation-case>
</navigation-rule>
```

NAVEGACIÓN IMPLÍCITA

JSF 2.x introduce una simplificación que reduce la complejidad en la navegación. Introduce el concepto de navegación implícita. Si no hay ningún caso de navegación coincidente después de comprobar todas las reglas disponibles, el controlador de navegación comprueba que el resultado de la acción corresponde al identificador de una vista. Si se encuentra una vista coincidente de los resultados de acción , se navega de forma implícita a la vista encontrada.

NAVEGACION

NAVEGACIÓN IMPLÍCITA

En el siguiente ejemplo vemos como en el **h:commandButton**, en el atributo **action**, indicamos una cadena.

```
<h:commandButton  
    action="listTutorialsView"  
    value="Entrar" />
```

Esta no es EL, por lo que no estamos haciendo referencia a un backbean. Esta cadena correspondería con el "outcome" que serviría para determinar la regla de navegación a disparar. Pero como no hemos escrito ninguna regla de navegación ¿qué es lo que va a hacer JSF 2.x? Sencillo, simplemente se limitará a buscar una página con el mismo nombre y la extensión .xhtml.

Es decir, si en nuestro ejemplo hemos puesto `action="listTutorialsView"`, JSF 2.x intentará saltar a la vista `listTutorialsView.xhtml`.

DEMO



EVALUACIÓN INMEDIATA

- Evalúa la expresión inmediatamente cuando la página es preparada para ser mostrada.
- Las expresiones tienen la forma `${ expr }`.
- Son de solo lectura.
- Pueden estar en la parte estática de la página y como valores de atributos en una etiqueta de JSF
- Ejemplo:

```
<fmt:formatNumber value="${sessionScope.factura.total}"/>
```

EVALUACIÓN DIFERIDA

- Evalúa la expresión cuando sea necesario dentro del ciclo de vida de la página web.
- Las expresiones tienen la forma `#{ expr }`
- Pueden leer y actualizar propiedades de los beans (lectura/escritura)
- Solo pueden estar como valores de atributos de una etiqueta de JSF.
- Ejemplo:

```
<h:inputText id="name" value="#{cliente.nombre}" />
```


EXPRESIONES DE METODOS

- Las expresiones pueden invocar métodos sin parámetros:
- Ejemplo:

```
<h:commandButton  
    id="procesar"  
    value="Procesar"  
    action="#{factura.procesar()}" />
```

EXPRESIONES DE METODOS

- Las expresiones pueden invocar métodos con parámetros.
- Ejemplos:

```
<h:commandButton  
    action="#{negocio.compra('TELEVISOR')}"  
    value="comprar"/>
```

DEMO



*¡ Muchas gracias
por su atención!*