



Manual de Plantillas para Listados

www.sagasoluciones.com

info@sagasoluciones.com

C/ Salado 11 Local 10. 41010 Sevilla

T. + 34 954 45 72 75

F. + 34 954 45 75 72

CONTROL DEL DOCUMENTO

Realizado

21/09/2010

Ignacio Villegas

Técnico

Revisado

24/09/2010

Sergio Raposo Vargas

Cargo

Control de Modificaciones

| <i>Revisión</i> | <i>Fecha</i> | <i>Autor/res</i> | <i>Descripción</i> |
|-----------------|--------------|------------------|-----------------------------------|
| Nº 1 | 22/09/2010 | Ignacio Villegas | Realización inicial del documento |

Índice de Contenido

| | | |
|----|--|----|
| 1. | INTRODUCCIÓN | 4 |
| 2. | CREACIÓN DEL RECUSO PLANTILLA DE LISTADO | 4 |
| 3. | CONFIGURACIÓN | 6 |
| 4. | RESUMEN DE LOS PATRONES | 17 |

1. Introducción

Para poder usar cualquier Recurso listado es necesario contar con una **plantilla de listado** que es la encargada de recoger los datos necesarios del recurso y pasarlos al listado para que se puedan mostrar.

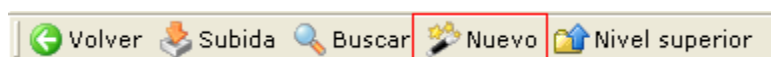
Hay varias plantillas listado creadas por defecto y configuradas para listar los recursos más comunes del Template Saga. Esas plantillas son las siguientes:

- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_generica_titulo.html
- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_generica_titulo_teaser.html
- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_textos.html
- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_noticias.html
- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_eventos.html
- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_enlaces.html
- /system/modules/com.saga.opencms.list/plantillas_listado_simple/plantilla_descarga.html


En este manual se explica cómo crear y configurar las plantillas listado

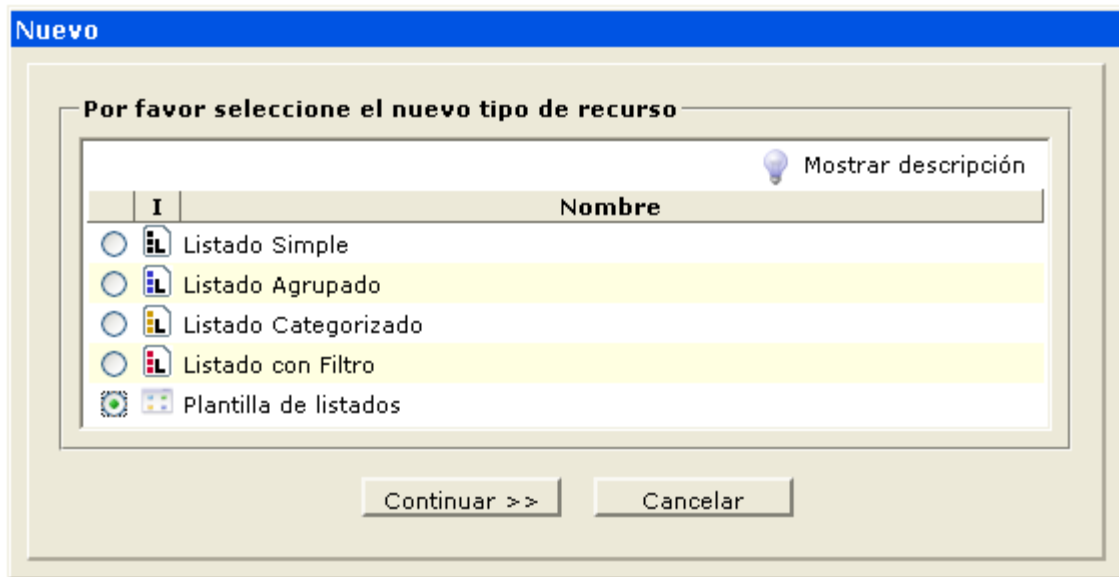
2. Creación del Recurso Plantilla de Listado

Para crear una Plantilla Listado pulsamos en el botón “Nuevo” de la barra de herramientas del OpenCms.



Nos aparece una ventana en la que tenemos que elegir la opción **[SAGA] Listados**

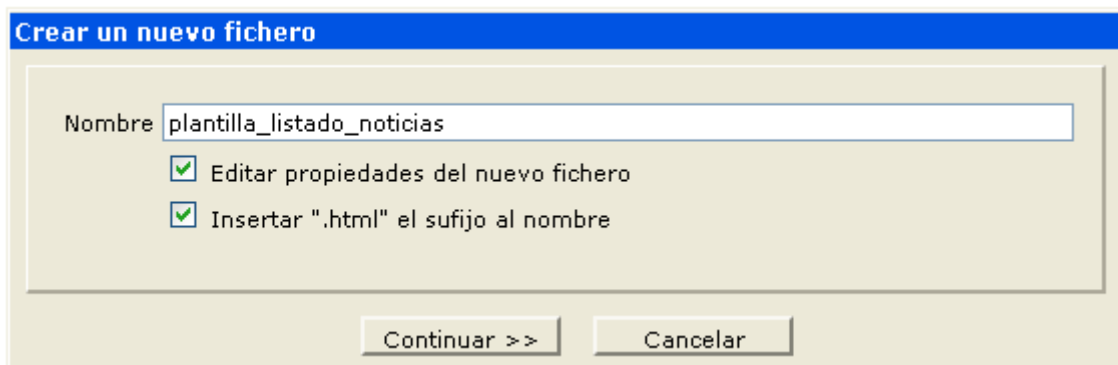
 [SAGA] Listados pulsamos continuar, y seguidamente nos aparecerá otra ventana donde seleccionamos el tipo de listado que queremos crear, en este caso seleccionamos el Recurso **Plantilla de listados**.



Pulsamos el botón continuar. Nos aparece una nueva ventana donde escribimos el nombre del recurso de listado.

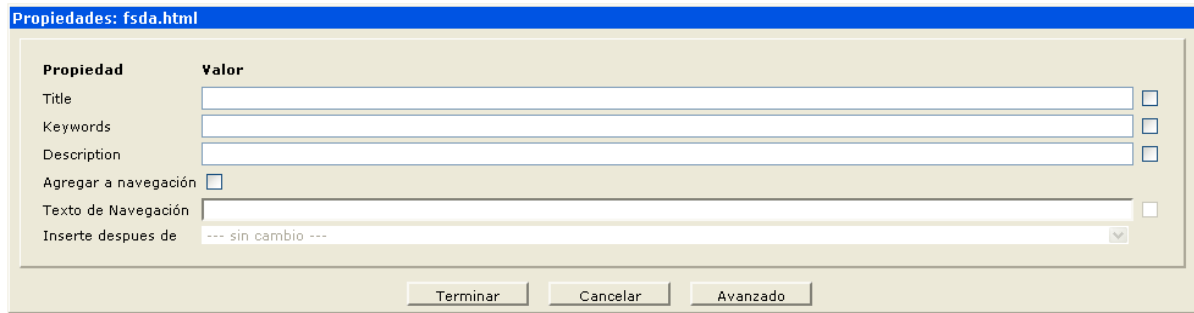
Nota importante: Es recomendable usar siempre la misma nomenclatura en el nombre de fichero. Este nombre debe ir sin acentos y sin espacios.

Ejemplo: listado-de-noticias.html



La casilla Insertar ".html" el sufijo al nombre, debe estar seleccionada.

Una vez introducido el nombre pulsamos “Continuar” y nos aparecerá una nueva ventana donde podemos editar las propiedades del recurso, podemos introducir un “Title” que corresponde con el Título del Recurso listado, el cual si pude llevar acentos y espacios y una descripción. Pulsamos Terminar.



| Propiedad | Valor |
|----------------------|--------------------------|
| Title | <input type="text"/> |
| Keywords | <input type="text"/> |
| Description | <input type="text"/> |
| Agregar a navegación | <input type="checkbox"/> |
| Texto de Navegación | <input type="text"/> |
| Inserte despues de | --- sin cambio --- |

Terminar Cancelar Avanzado

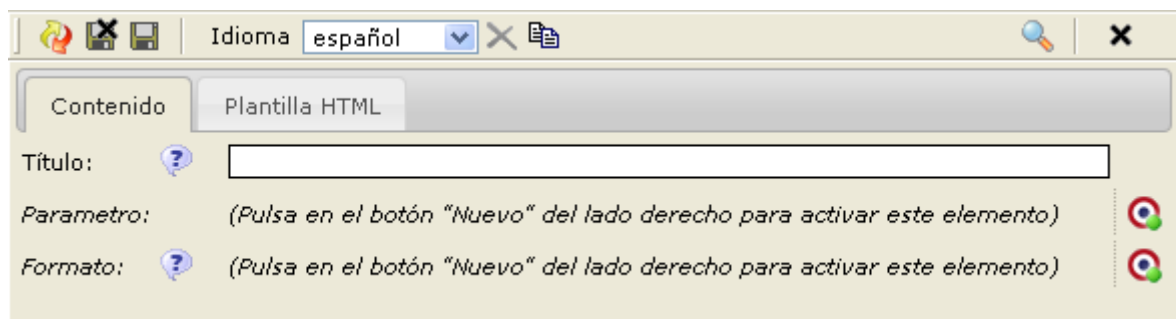
Una vez creado el recurso plantilla de listados pasamos a configurar el recurso.

3. Configuración

Para ello, nos posicionamos sobre el recurso que acabamos de crear y pulsamos el botón derecho y seleccionamos en el menú desplegable “editar” se nos abrirá una ventana donde demos introducir los parametros de configuración.

Veremos que contamos con dos pestañas: *Contenido*, *Plantilla HTML*

3.1. Contenido



Idioma: español

Contenido Plantilla HTML

Título:

Parametro: *(Pulsa en el botón "Nuevo" del lado derecho para activar este elemento)*

Formato: *(Pulsa en el botón "Nuevo" del lado derecho para activar este elemento)*

La pestaña *Contenido*, es donde a través de los parámetros recogemos los datos del recurso que queremos listar. Los campos que nos encontramos en esta pestaña son los siguientes:

Campo “**Título**” (*campo obligatorio*): Corresponde con el título de la plantilla listado.

Campo “**Parámetro**” (*campo obligatorio*): Cada recurso de OpenCms está formado por campos independientes, por lo que tenemos que hacer referencia a esos campos para que el listado pueda leer y recoger los valores del recurso a listar. Para eso usamos el campo “**Parámetro**”

Este campo a su vez está dividido en tres campos

- **Parámetro** (*campo obligatorio*): nombre del campo del recurso a mostrar. Dependerá del tipo de recurso. Por defecto debemos introducir el nombre del campo del recurso que queremos mostrar. Una buena forma de ver el nombre de los campos de los que está compuesto un recurso en concreto es pinchando con el botón derecho en el recurso que queramos (por ejemplos: noticias) y seleccionar en el menú desplegable la opción Avanzadas/Editar código de control, se abrirá un archivo xml, este archivo contiene la estructura XML del recurso y podemos ver que el contenido está dividido por bloques, donde cada bloque tiene un nombre de campo encerrado por <> ej.: <Title>. Este nombre es el que tenemos que introducir en el campo Parámetro del listado. Otra forma de saber la estructura del listado a mostrar es mirando su fichero de definición XSD. Ej: Title, Teaser, ...
Igualmente, los campos xml se pueden especificar mediante el prefijo: "xml:" pero este no es necesario, es decir, los siguientes valores tendrían el mismo efecto:
Title = xml:Title
También podemos mostrar una propiedad del recurso, para ello es necesario preceder al nombre de la propiedad la palabra "property:". Ej: Property:Title
- **Patrón** (*campo obligatorio*): Texto por el que se sustituirá el parámetro que acabamos de indicar arriba en la plantilla html, que debemos configurarla en la pestaña siguiente. El valor del patrón debe aparecer igual que en la Plantilla HTML. Los patrones serán por tanto remplazados por el valor del parámetro indicando en la casilla anterior, tanto por el valor del campo xml o el del property. En el caso que no exista ese campo, o no tenga valor, se sustituirá el patrón por un espacio en blanco. Ej: {TITULO} => Remplazará el patrón {TITULO} en la plantilla por el valor del campo Title del recurso tratado.

| Contenido | Plantilla HTML |
|----------------|--------------------|
| Título: | Plantilla Noticias |
| Parámetro: | Title |
| Patrón: | {TITULO} |
| Parámetro [1]: | Tipo: |
| | Texto |

| Contenido | Plantilla HTML |
|-----------|---|
| | <pre> <!--START_RECURSO_SGLG--> <li class="listadoli"> <p class="titulo_recurso">{TITULO} <p class="resumen_recurso">{TEASER}</p> <p class="resumen_recurso">Enlace de interés: {ENLACE}</p> <!--END_RECURSO_SGLG--> </pre> |

- **Tipo** (*campo obligatorio*): Indica el tipo del campo que se trate. Puede ser Texto, Imagen, Link, Descarga...
Es importante indicar siempre el tipo correcto del campo a mostrar, no es lo mismo querer mostrar una imagen que un recurso de tipo texto ya que el html que se genera es distinto en cada caso. Igualmente, según el tipo de contenido podremos aplicarle unos atributos u otro, por ejemplo, en el caso de un texto tiene sentido poder indicar el número máximo de caracteres a mostrar, y sin embargo, en el caso de una imagen, tendrá más sentido el ancho de la imagen.
Los posibles tipos que disponemos son:
 - Texto: Usado para parsear un simple campo de texto
 - Enlace: Este tipo nos permitirá introducir un enlace, el cual podremos personalizar utilizando los atributos propios.
 - Imagen: Este tipo nos permitirá introducir una imagen en nuestro listado, la cual podremos personalizar utilizando los atributos propios de las imagenes.

- Fecha: Este tipo se usará para parsear un recurso de tipo Fecha. El formato que se le aplicará será: dd/mm/aaaa (Ej: 24/09/2010)
 - Categoría: Lo usaremos cuando queramos mostrar la categoría a la que pertenece el recurso tratado.
 - Descarga: Aplica el mismo funcionamiento que el enlace.
 - Autonumérico: Un patrón de tipo autonumérico no necesita configurarle parámetro, y lo que hará es numerar nuestros recursos por orden de aparición, es decir, nos servirá por si queremos asignar un id distinto a cada
 - Campo dinámico: El campo dinámico nos permite realizar nuestras propios mapeos personalizados. Tan solo tendremos que programar una clase java que implemente la interfaz IDynamicBean
- **Atributos (campo opcional):**

Atributos del tipo imagen: Existen dos tipos de atributos, los atributos ATTR_ que son los propios de la etiqueta de html, y los atributos SCALE_ que son los propios del tag <cms:img> para escalar las imágenes.

Para asignar un valor del XML del recurso a los atributos de tipo ATTR_, se añade el símbolo “=” precedido de el nombre del campo, en caso contrario se puede escribir un valor constante sin especificarle ninguna marca de inicio.

Tipos de atributos ATTR:

- ATTR_ALIGN: Permite añadir a la etiqueta imagen el atributo “align” los valores pueden ser: ‘left’, ‘texttop’, ‘absmiddle’, ‘center’, ‘middle’, ‘bottom’, ‘right’.
- ATTR_ALT: Añade a la etiqueta imagen el atributo “alt”. Para recoger el valor desde el xml del recursos introducimos un ‘=’ y el valor del campo.
- ATTR_TITLE: Añade a la etiqueta imagen el atributo “title”. Para recoger el valor desde el xml del recursos introducimos un ‘=’ y el valor del campo.
- ATTR_BORDER: Añade a la etiqueta imagen el atributo “border”. Debemos introducir un valor numérico para especificar el grosor del borde.
- ATTR_HSPACE: Añade a la etiqueta imagen el atributo “hspace”. Debemos introducir un valor numérico para especificar el espacio horizontal entre la imagen.
- ATTR_VSPACE: Añade a la etiqueta imagen el atributo “vspace”. Debemos introducir un valor numérico para especificar el espacio entre la imagen.
- ATTR_ID: Añade el “id” que le indiquemos a la etiqueta imagen.
- ATTR_LONGDESC: Añade a la etiqueta imagen el atributo “longDesc”.
- ATTR_NAME: Añade a la etiqueta imagen el atributo “name”. Donde podemos introducir un nombre para las imágenes.
- ATTR_STYLE: Añade a la etiqueta imagen el atributo “style” que nos permite introducir estilo css a las imágenes, por ejemplo: style="border: 5px solid rgb(102, 0, 0);"

Tipos de atributos SCALE:

- **SCALE_ATTR_COLOR:** Este atributo especifica el color de fondo de la imagen al aplicarle un tipo de escalado. Debemos de introducir en el campo un color en formato #000000.
- **SCALE_ATTR_FILTER:** Permite aplicar a la imagen un filtro, tipo escala de grises o de sombra. Para usar más de un filtro debemos separarlos con “:” El tipo de valor que podemos introducir son ‘grayscale’ ‘shadow’.
- **SCALE_ATTR_HEIGHT:** Define la altura de la imagen. Debemos introducir un valor numérico el cual especifica la altura.
- **SCALE_ATTR_WIDTH:** Define la anchura de la imagen. Debemos introducir un valor numérico el cual especifica la anchura.
- **SCALE_ATTR_POSITION:** Este valor controla el la posición de la imagen dentro de los valores de escalado que le hemos asignado.
Los valores que podemos asignar son: : ‘0’: centro, ‘1’: abajo izquierda, ‘2’: abajo derecha, ‘3’: abajo medio, ‘4’: izquierda medio, ‘5’:derecha medio, ‘6’: arriba medio, ‘7’: arriba izquierda , ‘8’: arriba derecha
- **SCALE_ATTR_QUALITY:** Especifica la calidad de la imagen. Los valores permitidos son de 10 al 100, donde a mayor numero menos compresión de imagen y mayor calidad. Solo es aplicable para imágenes en formato jpg
- **SCALE_ATTR_TYPE:** Especifica el comportamiento que tendrá la imagen al aplicarle valores de ancho y alto al escalar la imagen.
Los valores que se pueden introducir son del 0 al 4 y donde el ‘0’ escala la imagen al tamaño exacto que se especifique, el ‘1’ crea un color de fondo al escalar la imagen y mantiene las proporciones de la imagen, el ‘2’ escala la imagen a tamaño que se especifica y recorta la el trozo de imagen para evitar que la imagen se deforme, el 3 escala y mantiene las proporciones de la imagen, y el 4 no escala la imagen usa los valores que tenga la imagen por defecto.

Atributos del tipo enlace:

- **ENLACE_href:** Podemos indicar la url de destino del enlace, en el caso de ser parte del contenido xml indicaremos el nombre del campo donde se encuentra la ruta. En caso de no indicarse se cogerá directamente el valor del campo Parámetro comentado con anterioridad.
- **ENLACE_target:** Indica donde se abrirá el enlace. “_blank” para nueva ventana o “_self” para abrirla en la misma. Por defecto siempre se abrirá en la misma ventana.
- **ENLACE_ descripción:** Texto del enlace a mostrar por defecto.
- **ENLACE_campo:** Campo del que se recogerá el texto a mostrar en el enlace. Si se rellena este atributo se ignorara en Enlace_ descripción.
Si ninguna de los atributos son rellenos, aparecerá como texto la uri del enlace y el target vació.

Atributos del tipo texto:

- **TEXTO_MAX_CARACTERES:** Indica el numero de caracteres que se mostraran, añadiendo puntos suspensivos en caso de que la oración continúe.

- Decorador de Texto: El atributo decorador es un atributo que nos permite modificar el texto que vamos a mostrar sin necesidad de modificar nuestra plantilla HTML. Para que funcione, usaremos el carácter especial “@” para simbolizar el contenido del campo indicado, a partir de ahí podremos agregarle texto o código html por delante o por detrás. La ventaja de usar esta decoración por encima de añadirlo en la plantilla radica en que, en el caso de no tener valor el campo, si lo hemos puesto en la plantilla no podemos controlar que no se pinte el texto, en el caso de usar este atributo, si no existe el campo no se mostrará nada.

Ej: parámetro = email; atributo decorador: escribe un correo a:

Si existe el campo email pintaría: escribe un correo

a:sergio.raposo@sagasoluciones.com; en caso contrario no pintaría nada.

Atributos del tipo campo dinámico:

- CLASE IDINAMIC BEAN: Indica la clase que debe generar el valor del campo. Para poder implementar un mapeo a medida hay que hacer una clase java que implemente la interfaz: *IDinamicBean*

Ejemplo: si estamos haciendo un listado de noticias y queremos listar el título de cada Noticia, tenemos que introducir en el campo parámetro Title , como se ve reflejado en le xml del recurso noticia. En el campo patrón introducimos {TITLE} el cual debemos referenciarlo en la plantilla HTML. En tipo introducimos Texto y en Atributos no es necesario introducir nada, pero podemos introducir un número máximo de caracteres usando el atributo TEXTO_MAX_CARACTERES e indicándole un número entero.



The screenshot shows a configuration window for a dynamic field. The 'Contenido' tab is selected, and the field is named 'Plantilla Noticias'. The configuration includes:

- Parametro:** Title
- Patrón:** {TITLE}
- Tipo:** Texto
- Atributo:** TEXTO_MAX_CARACTERES
- Valor:** 200

3.2. Plantilla HTML

Una vez indicado todos los parametros que queremos mostrar del recurso en el listado pasamos a configurar la pestaña **Plantilla HTML**.

Para crear una plantilla HTML es recomendable tener un conocimiento básico de html.

La plantilla HTML es la encargada de pintar los datos recogidos del recurso por pantalla. Para eso la plantilla nos ofrece la libertad de mostrar los datos con la estructura que queramos.

La plantilla esta formada por bloques. Según el tipo de listado que estemos usando podemos usar unos bloques u otros. Los bloques que pueden usarse en una plantilla son:

BODY_SGLG
MENU_SGLG (solo para los listados Agrupados)
MENU_ITEM (solo para los listados Agrupados)
GRUPO_SGLG (solo para los listados Agrupados)
RECURSO_SGLG
PAGINACION_SGLG
SEPARADOR_SGLG
NO_INFO_SGLG
INFORME_RESULTADOS_SGLG

Todos los bloques los podemos usar en los cuatro tipos de listado a excepción de los bloques **MENU_SGLG**, **MENU_ITEM_SGLG** y **GRUPO_SGLG** que solo son necesarios en los listados agrupados.

A) BODY_SGLG

Esta bloque es obligatorio y representa al cuerpo del listado, es decir, es el bloque principal que dará forma al listado. Se define usando las siguientes etiquetas

```
<!-- START_BODY_SGLG -->
<!--END_BODY_SGLG -->
```

Puede contener los siguientes patrones:

{SGLG_TITULO} Muestra el Título del listado.
{SGLG_TEASER} Muestra el Resumen del listado.
{SGLG_CLASE} Muestra la Clase general del listado configurada en el recurso.
{SGLG_ID} Muestra el Id del listado. Es importante asignar el id al listado para distinguirlos en caso que haya dos listados en la misma pantalla, por ejemplo, en la home.
{SGLG_TEXTOCABECERA} Muestra el Texto de cabecera que se especifica en el recurso listado
{SGLG_TEXTOPIE} Muestra el Texto del pie que se especifica en el recurso listado
{SGLG_LIST} Lugar donde se mostrará el listado. El html correspondiente al listado lo extraeremos del bloque RECURSO_SGLG
{SGLG_PAGINACION} Muestra la paginación del listado
{SGLG_MENU} Permite mostrar el Menu de los listados Agrupados
{SGLG_BLOQUES} Permite mostrar los bloques de los listados Agrupados

Para los listados de tipo Listado con Filtro podemos usar el patron **{FILTRO}** el cual mostraría el filtro que hemos aplicamos al listado. En el caso que un listado no sea de tipo filtrado, se remplazará por un espacio en blanco para que no se vea afectado el resto de tipos de listado.

Ejemplo:

```

<!--START_BODY_SGLG-->
<div class="{SGLG_CLASE}">
<div class="titulo_listado">
  <h1>{SGLG_TITULO}</h1>
</div>
  <div class="texto_cabecera">
    {SGLG_TEXTOCABECERA}
  </div>
  {FILTRO}
  <ul>
    {SGLG_LIST}
  </ul>
  {SGLG_PAGINACION}
  <div class="texto_pie">
    {SGLG_TEXTOPIE}
  </div>
</div>
<!--END_BODY_SGLG-->

```

B) MENU_SGLG (solo para los listados Agrupados)

Esta bloque es obligatorio. Se define usando las siguientes etiquetas

```

<!-- START_MENU_SGLG-->
<!--END_MENU_SGLG -->

```

Para poder mostrar este bloque es necesario que en el bloque BODY_SGLG contenga el patrón {SGLG_MENU}

Este bloque es usado por los listados Agrupados y permite formar el menu de cada grupo del listado.

Puede contener los siguientes patrones:

{SGLG_MENUITEMS} Muestra cada item del menú

Ejemplo:

```

<!--START_MENU_SGLG-->
<div class="menuListado">
  <ul class="sglgMenu">{SGLG_MENUITEMS}</ul>
</div>
<!--END_MENU_SGLG-->

```

C) MENU_ITEM (solo para los listados Agrupados)

Esta bloque es obligatorio. Se define usando las siguientes etiquetas

```
<!-- START_MENU_ITEM_SGLG -->
<!--END_MENU_ITEM_SGLG -->
```

Para poder mostrar este bloque es necesario que en el bloque MENU_SGLG contenga el patrón { SGLG_MENUITEMS }

Puede contener los siguientes patrones:

- {SGLG_ENLACEBLOQUE}: Permite obtener el enlace en cada bloque
- {SGLG_TITULOBLOQUE}: Muestra el título del cada Bloque
- {SGLG_NUMELEMENTOSBLOQUE}: Muestra el número de elementos que contiene el grupo actual.

Ejemplo:

```
<!--START_MENU_ITEM_SGLG-->
<li><a href="{SGLG_ENLACEBLOQUE}">{SGLG_TITULOBLOQUE}</a></li>
<!--END_MENU_ITEM_SGLG-->
```

D) GRUPO_SGLG (solo para los listados Agrupados)

Este bloque es obligatorio y permite mostrar los bloques de los listados agrupados. Se define usando las siguientes etiquetas

```
<!-- START_GRUPO_SGLG -->
<!--END_GRUPO_SGLG-->
```

Puede contener los siguientes patrones:

- {SGLG_ENLACEBLOQUE}: Permite obtener el enlace en cada bloque
- {SGLG_TITULOBLOQUE}: Muestra el título del cada Bloque
- {SGLG_NUMELEMENTOSBLOQUE}: Muestra el número de elementos que contiene el grupo actual.
- {SGLG_RECURSOS} Muestra los recursos a listar dentro de cada bloque. Equivalente al patron {SGLG_LIST} que usamos en el resto de plantillas de listado.

Ejemplo:

```
<!--START_GRUPO_SGLG-->
<div class="bloque">
<div class="tituloBloqueListado">
<a href="{SGLG_ENLACEBLOQUE}">{SGLG_TITULOBLOQUE}</a>
</div>
<ul class="recursosBloqueListado">{SGLG_RECURSOS}</ul>
```

```
</div>
<!--END_GRUPO_SGLG-->
```

E) RECURSO_SGLG

Este bloque es obligatorio y permite mostrar cada uno de los campos del recurso que estemos listando.

Para poder mostrar este bloque es necesario que en el bloque GRUPO_SGLG contenga el patrón {SGLG_RECURSOS}, el cual definimos como se indica abajo.

Se define usando las siguientes etiquetas:

```
<!--START_RECURSO_SGLG-->
<!--END_RECURSO_SGLG-->
```

Puede contener los siguientes patrones:

- {PAR-IMPAR}: Pintará “par” o “impar” en caso de tratarse un registro de número par o impar. Útil para cuando necesitamos hacer un listado donde se pinte cada registro de un color.
- {uri}: Pinta la url del recurso tratado, necesario para montar el enlace al detalle del recurso, por ejemplo, en el listado de una noticia, pondremos el título como un enlace cuyo campo href será {uri}. De esta forma al pinchar sobre el título nos iremos al detalle de la noticia.
- {SGLG_NUMERO_COMENTARIOS}: Pinta el número de comentarios que se han escrito sobre el recurso tratado. Útil por ejemplo en listados de artículos de un blog.

Y todos los patrones que hemos definido en la pestaña de Contenido como por ejemplo el {TITLE}, {TEASER}, {IMAGE}, o los que cada uno haya estimado oportuno.

Ejemplo:

```
<!--START_RECURSO_SGLG-->
  <li class="{PAR-IMPAR}">
    <span class="imagen">{IMAGEN}</span>
    <p class="titulo"><a href="{uri}">{TITLE}</a></p>
    <p class="fecha">{FECHA}</p>
    <p class="teaser">{TEASER}</p>
  </li>
<!--END_RECURSO_SGLG-->
```

F) PAGINACION_SGLG

Esta bloque es opcional y permite mostrar la paginación si se encuentra activada la opción de paginación en el listado. Si no se encuentra patron se generara la paginación por defecto.

Se define usando las siguientes etiquetas:

```
<!--START_PAGINACION_SGLG -->  
<!--END_PAGINACION_SGLG-->
```

Los patrones que se le pueden aplicar son los siguientes:

{PRIMERA_PAGINA}: indicara la ruta hacia la primera página.

{ANTERIOR_PAGINA}: indicara la ruta hacia la página anterior.

{SIGUIENTE_PAGINA}: indicara la ruta hacia la página siguiente.

{ULTIMA_PAGINA}: indicara la ruta hacia la última página.

{PA} {/PA}: El html contenido entre estas etiquetas se ocultara para la pagina 1. Obligatorio si aparecen los patrones {PRIMERA_PAGINA} o { ANTERIOR_PAGINA }

{PAGINACION_CENTRAL}: Mostrara el listado numérico de páginas.

{UA} {/UA}: El html contenido entre estas etiquetas se ocultara para la última página. Obligatorio si aparecen los patrones { ULTIMA_PAGINA } o { SIGUIENTE_PAGINA }

{INFORME_RESULTADO} Muestra el informe de resultados obtenidos.

G) SEPARADOR_SGLG

Esta bloque es opcional. Se utilizará para separar el listado numérico de la {PAGINACION_CENTRAL}.

Se define usando las siguientes etiquetas:

```
<!--START_SEPARADOR_SGLG -->  
<!--END_SEPARADOR_SGLG -->
```

Por defecto se toma “-”.

Se defina de la siguiente manera:

```
<!--START_TEMPLATE_SEPARADOR-->  
,  
<!--END_TEMPLATE_SEPARADOR-->
```

Donde “,” sería el separador.

H) NO_INFO_SGLG

Esta bloque es opcional. Con este patrón podremos definir el texto que aparecerá en el caso que no haya información que mostrar, es decir, que el listado esté vacío. Por defecto el texto que se mostrará es: “No hay información”

Se define usando las siguientes etiquetas:

```
<!--START_NO_INFO_SGLG-->  
<!--END_NO_INFO_SGLG-->
```

Ejemplo:

```
<!--START_NO_INFO_SGLG-->  
No se ha encontrado información que mostrar.  
<!--END_NO_INFO_SGLG-->
```

I) INFORME_RESULTADOS_SGLG

Esta bloque es opcional. Se utilizará para cambiar el texto del Informe de resultados del listado, que es el que nos indica el número de elementos que estamos con respecto al total.

Se define usando las siguientes etiquetas:

```
<!--START_INFORME_RESULTADOS_SGLG-->  
<!--END_INFORME_RESULTADOS_SGLG-->
```

Los patrones que se le pueden aplicar son los siguientes:

{TOTAL_REGISTROS} Muestra el total de recursos encontrados.
{PRIMER_REGISTRO} Muestra el primer registro que se está visualizando según la página actual
{ULTIMO_REGISTRO} Muestra el último registro que se está visualizando según la página actual.

Este bloque se pintará donde le indiquemos en el caso de personalizar la plantilla de paginación. En caso contrario se pintará por defecto por debajo de la paginación.

En el caso que no configuremos este bloque, por defecto se mostrará un texto tal que:

“Mostrando resultados del 1 al 10 de 200 registros encontrados”

4. Resumen de los Patrones

| BLOQUES |
|---|
| <!--START_BODY_SGLG--> ... <!--END_BODY_SGLG--> |
| <!--START_MENU_SGLG --> ... <!--END_MENU_SGLG --> |
| <!--START_MENU_ITEM_SGLG--> ... <!--END_MENU_ITEM_SGLG --> |
| <!--START GRUPO_SGLG --> ... <!--END GRUPO_SGLG --> |
| <!--START_RECURSO_SGLG--> ... <!--END_RECURSO_SGLG --> |
| <!--START_PAGINACION_SGLG--> ... <!--END_PAGINACION_SGLG --> |
| <!--START_SEPARADOR_SGLG--> ... <!--END_SEPARADOR_SGLG --> |
| <!--START START_NO_INFO_SGLG --> ... <!--END START_NO_INFO_SGLG --> |
| <!--START_INFORME_RESULTADOS_SGLG--> ... <!--END_INFORME_RESULTADOS_SGLG--> |

| PATRONES | DESCRIPCIÓN |
|----------------------|---|
| {SGLG_TITULO} | Título del listado |
| {SGLG_TEASER} | Resumen del listado |
| {SGLG_CLASE} | Clase general del listado configurada en el recurso |
| {SGLG_ID} | Id del listado |
| {SGLG_TEXTOCABECERA} | Texto de cabecera |
| {SGLG_TEXTOPIE} | Texto del pie |
| {SGLG_LIST} | Lugar donde se mostrará el listado |
| {SGLG_MENU} | Menú en el caso de los agrupados |
| {SGLG_MENUITEMS} | Elementos del menú en el caso de disponer de ellos |
| {SGLG_PAGINACION} | Lugar donde mostraremos la paginación |
| {SGLG_ENLACEBLOQUE} | Enlace al bloque |

| | |
|---------------------------|--|
| {SGLG_TITULOBLOQUE} | Título del bloque en el listado agrupado |
| {SGLG_NUMELEMENTOSBLOQUE} | Muestra el número de elementos dentro del grupo tratado |
| {SGLG_NUMERO_COMENTARIOS} | Muestra el número de comentarios del recurso |
| {SGLG_RECURSOS} | Recursos dentro del bloque |
| {START_IMG} | Abre el bloque de la imagen |
| {END_IMG} | Cierra el bloque de la imagen |
| {uri} | Url del recurso tratado |
| {SGLG_URI_LISTADO} | Url del listado |
| {PRIMERA_PAGINA} | Link a primera página |
| {ANTERIOR_PAGINA} | Link a página anterior |
| {SIGUIENTE_PAGINA} | Link a siguiente página |
| {ULTIMA_PAGINA} | Link a última página |
| {PAGINACION_CENTRAL} | Números de la paginación |
| {PA} | Patrón de apertura del bloque de retroceder página |
| {/PA} | Patrón de cierre del bloque de retroceder página |
| {UA} | Patrón de apertura del bloque de avanzar página |
| {/UA} | Patrón de cierre del bloque de avanzar página |
| {PAR-IMPAR} | Clase par o impar según el número del registro mostrado |
| {PRIMER_REGISTRO} | Pinta el primer registro encontrado en la página actual |
| {ULTIMO_REGISTRO} | Pinta el último registro encontrado en la página actual |
| {TOTAL_REGISTROS} | Pinta el total de recursos encontrados |
| {INFORME_RESULTADO} | Pinta el informe de resultados obtenidos |
| {FILTRO} | Pinta la caja de filtrado para los listados de tipo filtrados |
| {SGLG_NUMERO_COMENTARIOS} | Muestra el número de comentarios que hay en los recursos listados. |