

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



» Estás en: [Inicio](#) [Tutoriales](#) ZK 6.5 Consejos para dispositivos móviles



Francisco Ferri Pérez

Consultor tecnológico de desarrollo de proyectos informáticos.

Desarrollador de proyectos informáticos, Microsoft Certified IT Professional - Enterprise Administrator

[Ver todos los tutoriales del autor](#)



Catálogo de servicios Autentia



Fecha de publicación del tutorial: 2012-12-05

Tutorial visitado 1 veces [Descargar en PDF](#)

Framework ZK



ZK 6.5 Consejos para dispositivos móviles

20 de septiembre de 2012, artículo original de [Monty Pan](#), Engineer, Potix Corporation, versión de ZK 6.5

Contenido

1. Introducción
2. Diferentes estilos
 1. No se soportan algunos Mold (aspectos gráficos y de comportamiento)
3. Diferentes eventos
 1. Eventos del ratón
 2. ClientInfoEvent
4. Otras consideraciones acerca del scrolling
5. Tan simple como sea posible
6. Conclusión
7. Referencias

Introducción

Este tutorial pertenece a la siguiente serie:

1. [ZK Responsive Design, enfoque/filosofía](#)
2. [ZK 6.5 Responsive design](#)
3. [ZK 6.5 Empezando con el Responsive Design](#)
4. [ZK 6.5 Consejos para dispositivos móviles](#)

Desde la versión 6.5 de ZK, los desarrolladores pueden aprovechar el poder de ZK para crear una aplicación web sin problemas en dispositivos móviles como una tablet.

Tal y como tiene que ser, puesto que existen diferencias entre el hardware de una tableta y un PC, debemos pensar diferente en la fase de diseño, teniendo en cuenta cada detalle que necesita ser ajustado para que se vea como esperamos en todos los dispositivos, y por lo tanto consigamos proveer de una buena experiencia al usuario en ambos.

El siguiente artículo está basado en ZK 6.5, explorando y discutiendo cómo puedes controlar y usar ZK 6.5 para hacer el mejor uso de él.

Diferentes estilos

En una tableta, los usuarios operan con sus dedos, mediante interacciones de tipo táctiles, por lo tanto el tamaño de los componentes en los que tiene que clicar (por ejemplo botones) no puede ser muy pequeño, 24px x 24px es el límite.

Por otra parte, si el tamaño del componente es pequeño, el espacio entre componentes debe ser grande, para evitar que el usuario pulse los botones cercanos por error.

ZK ha ajustado los componentes de acuerdo a este principio, por ejemplo `combobox`, `timebox` y `colorbox`, los desarrolladores ahora pueden usarlos en un dispositivo móvil directamente. Para más detalles, te recomendamos que leas: [UI Enhancements](#).

No se soportan algunos Mold (aspectos gráficos y de comportamiento)

Síguenos a través de:



Últimas Noticias

» ESTRENO de Terrakas 1x05: "Un lugar en el mundo"

» Mi retrospectiva sobre la CAS 2012

» Autentia estuvo en el Duatlón de Torrejón de Ardoz

» Participamos en la Carrera de las Empresas 2012

» ¡¡¡Terrakas 1x04 recién salido del horno!!!

[Histórico de noticias](#)

Últimos Tutoriales

» Inserción de gráficas JFreeChart en un documento Excel mediante la librería POI de Apache

» ZK 6.5 Empezando con el Responsive Design

» Cálculo de una nómina

» ZK 6.5 Responsive design

» [LoadUI: pruebas de rendimiento de servicios web.](#)

Últimos Tutoriales del

En aplicaciones web destinadas a navegadores clásicos de escritorio, en algunos componentes se disponía de diferentes aspectos gráficos y de comportamientos, para cambiar su aspecto y comportamiento en función de este atributo, pero por diferentes causas operacionales y de no ser aplicable a los dispositivos móviles, ZK ya no soporta el atributo `Mold` en dispositivos móviles. Para una lista detallada al respecto te recomendamos que leas [Unsupported Molds](#).

A continuación te pongo algunos casos respecto del atributo `Mold` no soportado:

- Componentes relacionados con la entrada de datos: No se soporta el `Mold rounded`.
- `Button`: no soporta los `mold os` y `trendy`.
- `GroupBox`: no soporta los `mold default` se usará `sd` como el de por defecto.
- `Splitter`: no se soporta el `mold os`.
- `Tabbox`: no se soporta el `mold accordion-lite`.

Diferentes eventos

Eventos de Ratón

Como no hay ratón o cursor en las tabletas, los eventos relacionados con el ratón son diferentes que en el escritorio tradicional.

Primero de todo, `onmouseover` no está soportado, `tooltip` y `autodrop` tampoco funcionan en una tablet. Segundo, `onrightclick` no está soportado directamente por el browser nativo como evento, pero **ZK simula este evento del siguiente modo "pulsar y aguantar pulsado un momento"**, por lo tanto la propiedad `context` (que muestra el menú contextual) y `onrightclick` pueden ser usados en una tablet.

Por otra parte, si a `Image` o `Div` se le añade la propiedad `onclick`, los desarrolladores deben añadir el CSS `cursor: pointer`; en el atributo `class` o `style`, para que el navegador del tablet pueda gestionar la propiedad `onclick` correctamente.

ClientInfoEvent

Los usuarios pueden cambiar fácilmente la orientación de la tablet, por lo tanto los desarrolladores tienen que poder identificar cual es la orientación que los usuarios están usando y aplicar los cambios necesarios al tamaño y diseño. ZK 6.5 añade esta información sobre la orientación en el objeto `ClientInfoEvent`, simplemente añadimos atributo de evento `onClientInfo` para obtener el objeto `ClientInfoEvent` y accedemos a su método `getOrientation()`, `isPortrait()` o `isLandscape()`.

Por ejemplo:

```

view plain print ?
01. <tabbox id="tbx" height="400px" width="600px">
02.
03.     <attribute name="onClientInfo"><!--[CDATA[
04.         ClientInfoEvent oe = (ClientInfoEvent) event;
05.         lbl.setValue(oe.getOrientation());
06.         if (oe.isPortrait()) {
07.             tbx.setHeight("600px");
08.             tbx.setWidth("400px");
09.         } else {
10.             tbx.setHeight("400px");
11.             tbx.setWidth("600px");
12.         }
13.     ]--></attribute>
14.
15.     <tabs>
16.         <tab label="tab 1" />
17.     </tabs>
18.     <tabpaneles>
19.         <tabpanel>
20.             Current Orientation:
21.             <label id="lbl" />
22.         </tabpanel>
23.     </tabpaneles>
24. </tabbox>

```

Otras consideraciones acerca del scrolling

En el artículo [Scrolling on Tablet](#) hablamos sobre el principio básico de scroll en tablets.

A continuación, discutiremos a cerca de algunos componentes muy comunes que puede que uses en tu implementación.

- Para crear un nuevo text area, puedes usar `<textbox multiline="true" />` cuando el contexto es grande, puedes hacer scroll en el componente `Textbox`. Este comportamiento lo maneja directamente el navegador, nada que ver con ZK, por lo tanto, su comportamiento será diferente de otros componentes en los que es ZK quien controla el scroll.
- Si el sistema contiene un `Image`, el desarrollador debe especificar el tamaño para o activar la precarga mediante el atributo a tal efecto, de cualquier otro modo, el scroll no funcionará como se espera. Veamos un ejemplo a continuación.

```

view plain print ?
01. <zk>
02.     <window contentstyle="overflow:auto" height="300px" border="normal">
03.         <image
04.             src="http://www.zkoss.org/resource/img/index/src/top_bannerimage3.png" />
05.         <div>bottom</div>
06.     </window>
07. </zk>

```

En este caso, `Window` no producirá una barra de scroll, el usuario final de la aplicación no podrá ver la imagen entera, puesto que no podrá hacer scroll de la misma.

Esto se soluciona de 2 formas, 1. Activando el atributo de `preload` como en el siguiente ejemplo.

```

view plain print ?
01. <zk>
02.     <window contentstyle="overflow:auto" height="300px" border="normal">

```

Autor

» [ZK 6.5 Empezando con el Responsive Design](#)

» [ZK 6.5 Responsive design](#)

» [ZK Responsive Design, enfoque/filosofía](#)

» [Tutorial de iniciación en Framework ZK](#)

» [Tu aplicación web ZK enfoque MVVM \(5-5\)](#)

Últimas ofertas de empleo

2011-09-08

 [Comercial - Ventas - MADRID.](#)

2011-09-03

 [Comercial - Ventas - VALENCIA.](#)

2011-08-19

 [Comercial - Compras - ALICANTE.](#)

2011-07-12

 [Otras Sin catalogar - MADRID.](#)

2011-07-06

 [Otras Sin catalogar - LUGO.](#)

```

03. <custom-attributes org.zkoss.zul.image.preload="true"/>
04. <image
05.   src="http://www.zkoss.org/resource/img/index/src/top_bannerimage3.png" />
06. </div>bottom</div>
07. </window>
08. </zk>

```

2. O indicando el tamaño de la imagen, como en el siguiente ejemplo:

```

view plain print ?
01. <zk>
02. <window contentstyle="overflow:auto" height="300px" border="normal">
03.   <image height="500px"
04.     src="http://www.zkoss.org/resource/img/index/src/top_bannerimage3.png" />
05.   </div>bottom</div>
06. </window>
07. </zk>

```

Nuestra recomendación es activar el atributo de precarga en el fichero de configuración zk.xml para evitar tener que hacerlo en todas las imágenes, una a una.

- No te recomendamos que asignes el número de filas en el componente `Listbox`, es decir que no establezcas ningún valor en el atributo `rows`, como en el siguiente ejemplo.

```

view plain print ?
01. <zk>
02.   <zscript> String[] s = new String[100]; </zscript>
03.   <listbox id="lbx1" height="300px">
04.     <zk foreach="{s}">
05.       <listitem label="{forEachStatus.index + 1}" />
06.     </zk>
07.   </listbox>
08.   <listbox id="lbx2" rows="8">
09.     <zk foreach="{s}">
10.       <listitem label="{forEachStatus.index + 1}" />
11.     </zk>
12.   </listbox>
13. </zk>

```

En el iPad, solo una pequeña parte de los 10 elementos del componente `lbx1` es visible, el usuario final verá que es incompleto y hará scroll de forma natural. Este efecto no es el mismo en el componente `lbx2`, puesto que hemos establecido el número de filas "rows" a un valor. Esto sucede de igual modo en los componentes `Grid` y `Tree`.

Tan simple como sea posible

Como hemos mencionado antes, el tamaño de la pantalla en tablets o teléfonos móviles es mucho más pequeño y los componentes crecen respectivamente para facilitar al usuario final la interacción con ellos. Por lo tanto, no es prudente crear un diseño complejo para que los usuarios interactúen con él meticulosamente.

La potencia de procesar es mucho menor en un tablet que en un PC tradicional. El tiempo de renderizado en un ordenador de escritorio, puede ser lo suficientemente breve como para ignorarlo, pero puede ser un problema en una tablet. Por lo tanto el diseño de websites para tableta siguiendo el concepto "Tan simple como sea posible" es muy importante.

Otra forma de mejorar la velocidad de rendimiento es activar el atributo `ROD` de los componentes de tipo `Listbox`, `Grid` y `Tree`.

Conclusión

ZK 6.5 provee de la habilidad de crear aplicaciones web para el escritorio y la tableta desde el mismo código de base. Pero teniendo en cuenta a la hora de desarrollar algunas de las diferencias a las que hemos apuntando. Siguiendo este artículo, el desarrollador puede evitar estas consideraciones y crear aplicaciones web de forma muy productiva.

A disfrutar de ZK 6.5!

Referencias

- [Página web oficial de ZK](#)

Este documento es un extracto de la documentación oficial del Framework ZK, traducido y ampliado por Francisco Ferri. Colaborador de Potix (creadores del Framework ZK). Si quieres contactar con él puedes hacerlo en franferri@gmail.com, en twitter [@franciscoferri](#) o en [LinkedIn](#)

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

Share |

0

¡Anímate y coméntanos lo que pienses sobre este **TUTORIAL**!



» **Regístrate** y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

IMPULSA

Impulsores

Comunidad

¿Ayuda?

sin clicks

0 personas han traído clicks a esta página



powered by [kamacracy](#)

Copyright 2003-2012 © All Rights Reserved | [Textos](#) | [Legal](#) | [Contacto](#) | [Condiciones de uso](#) | [Banners](#) | Powered by Autentia

