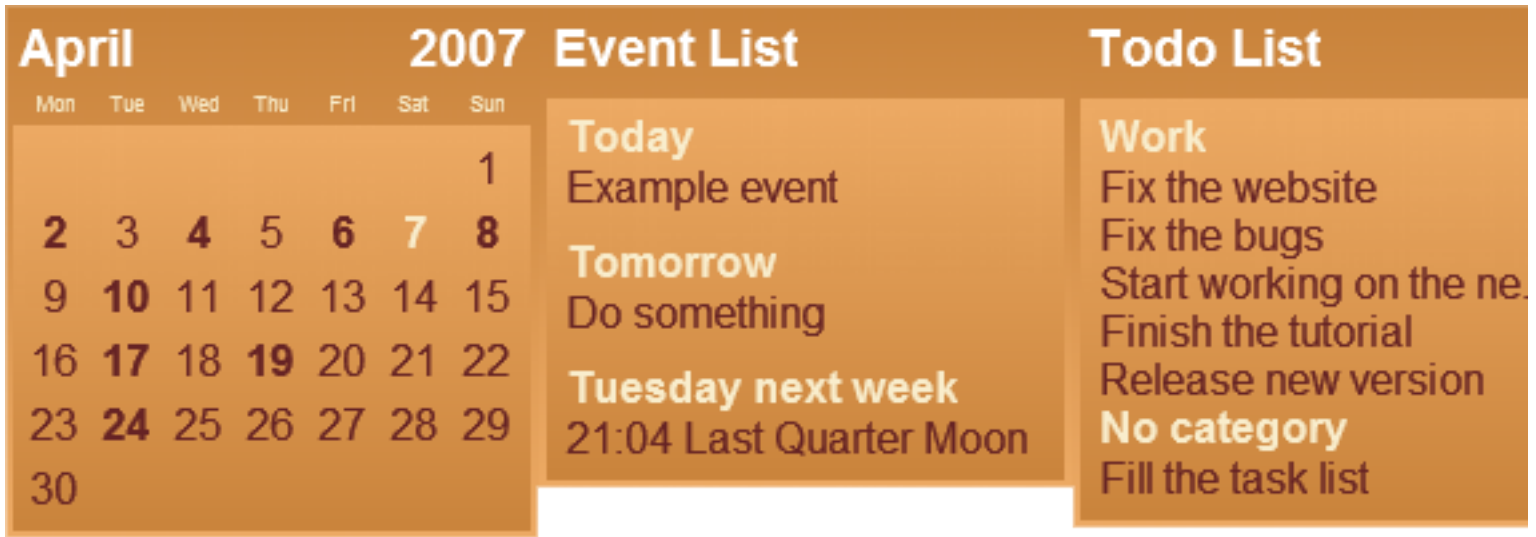


Tutorial de Temas



En este tutorial crearemos un tema de calendario muy simple desde cero. El tema no usará imágenes por lo que debería ser fácil de modificar. Deberías tener algunos conocimientos básicos de xml para poder seguir este tutorial.

Antes de empezar a crear temas debes asegurarte de tener un editor de texto que soporte la codificación utf-8. No edites los archivos con Notepad! Consigue un editor como [PSPad](#) en cambio. En Linux gedit o kate están bien.

Si estás tratando de editar un tema existente puedes cambiar la extensión del archivo .r2skin por .zip y descomprimir el archivo para acceder a los archivos xml y de imágenes.

Para empezar debemos crear una carpeta para nuestro nuevo tema. Así que ve a la subcarpeta de temas de Rainlendar en la carpeta de la aplicación y crea una nueva carpeta. La llamaremos *Nahka* (esto es "tema" en finlandés :-). Este será también el nombre de nuestro tema.

Dentro de la carpeta *Nahka* crea un nuevo archivo de texto. El nombre del archivo debe ser *skin.xml*. Rainlendar también es capaz de usar los temas xml en forma descomprimida lo que hace las cosas más fáciles durante la fase de desarrollo porque podemos refrescar el calendario para ver los cambios.

Ahora, abre el *skin.xml* con un editor de textos.

El archivo xml necesita una cabecera específica así que copia el texto que está debajo en el editor.

```
<?xml version="1.0" encoding="UTF-8"?>
```

La etiqueta principal en los temas de Rainlendar2 es <skin> la cual debe contener a las otras etiquetas. Entonces, agreguemoslas.

```
<skin version="1.0">
</skin>
```

Nota la version="1.0" en la etiqueta de apertura. Ello define la versión del formato r2skin de Rainlendar. Actualmente debería siempre estar establecido a "1.0" pero si en el futuro el formato cambia este valor es usado para determinar los detalles del formato. La version="1.0" en la etiqueta <?xml> es completamente diferente así que no lo cambies.

Todos los temas necesitan tener una sección de información que defina el nombre del tema y algunos otros detalles. Este es un ejemplo

```
<info>
  <version>1.0</version>
  <author>Rainy</author>
  <email>rainy@iki.fi</email>
  <homepage>http://www.rainlendar.net</homepage>
  <comment>The result of the skinning tutorial.</comment>
</info>
```

El elemento version contiene el número de versión del tema. Puedes incrementarlo cuando hagas cambios en el tema para que la gente que lo usa pueda saber que ha sido actualizado. Los elementos correo electrónico y página de inicio pueden dejarse en blanco si no quieres publicar ese tipo de información. El elemento comment puede contener p.ej. instrucciones de cómo usar el tema o lo que quieras poner allí.

Anterior - [Siguiete](#)

El skin.xml debería ser como lo siguiente (aún no funciona así que no es necesario probarlo):

```
<?xml version="1.0" encoding="UTF-8"?>
<skin version="1.0">
  <info>
    <version>1.0</version>
    <author>Rainy</author>
    <email>rainy@iki.fi</email>
    <homepage>http://www.rainlendar.net</homepage>
    <comment>The result of the skinning tutorial.</comment>
```

```
</info>  
</skin>
```

La Ventana Calendario

Antes de que empecemos a crear la ventana principal del calendario debemos definir los elementos gráficos. Estos incluyen las fuentes y los gráficos de fondo. Los elementos gráficos son definidos con la etiqueta `<element>`. Primero crearemos simplemente una fuente que es utilizada para dibujar los días en el calendario.

```
<elements>  
  <font id="font.normal" facename="Arial" size="12" />  
</elements>
```

El atributo *id* es la cadena de identificación que es utilizada cuando se hace referencia al elemento. Puede ser cualquier cadena y tan larga como quieras mientras no se repitan.

Luego crearemos la ventana para el calendario. Una ventana es simplemente un contenedor para los ítems. No conoce y no le importa lo que contenga por lo que no hay ventanas de calendario, ventanas de lista de eventos y así; son simplemente ventanas que contienen ítems. La ventana no está a contener un solo ítem. En este ejemplo en realidad tendrá múltiples ítems (calendario, año, mes y un par de imágenes) y también puede contener la lista de eventos y la lista de tareas.

Una ventana es definida con la etiqueta `<window>`.

```
<window id="Calendar" w="200" h="200" threshold="48" default="1">  
</window>
```

El *id* es nuevamente el identificador de la ventana. También define el nombre de la ventana que es mostrado en las opciones de los temas. El *w* y *h* son el ancho y la altura de la ventana. No siempre es necesario definirlos pues Rainlendar puede automáticamente calcular las dimensiones. Sin embargo esto no funciona a menos que los contenidos tengan un tamaño y posición específicos, así que a menos que estés haciendo una ventana que puede tener diferentes tamaños (p.ej. usando una lista de eventos o tareas) es útil definir las dimensiones exactas para la ventana.

El *threshold* es utilizado con la región de transparencia y define el valor (0-255) para los píxeles transparentes. Para definir un buen valor cambia la región de transparencia en las opciones avanzadas y observa que valor produce los mejores resultados.

Si el atributo *default* es establecido a 1 la ventana es mostrada por defecto cuando el

tema es seleccionado en el modo simple. Deberías establecerlo para todas las ventanas comunes (calendario, lista de eventos, lista de tareas).

Dentro de la ventana crearemos el calendario.

```
<calendar id="Calendar.calendar" x="0" y="0" w="200" h="200" layout="GRID" style="background-color:#f0f0f0">
  <days>
    <appearance layer="0" priority="0" element="font.normal" color="102,102,102" />
  </days>
  <today>
    <appearance layer="0" priority="100" element="font.normal" color="255,0,0" />
  </today>
  <events>
    <appearance layer="0" priority="1" element="font.normal" color="102,102,102" />
  </events>
</calendar>
```

El calendario está definido para abarcar la ventana entera por ahora. El *layout* define como aparecen los días en el calendario. El GRID es una tabla estándar de 7x6.

Los `<days>`, `<today>` y `<events>` definen como los días normales, el día actual y los eventos deberían verse. Usaremos la fuente, que fue definida en los `<elements>`, para todos ellos. Todas las apariencias utilizan la misma capa lo que significa que solo aquella que tenga la mayor prioridad será dibujada en un día en particular. En nuestro caso el día actual tiene muy alta prioridad por lo que es dibujada sin importar que. Si un día tiene un evento su apariencia es dibujada en el calendario y si no el día mostrará simplemente el número de un día normal. La prioridad del evento es ajustada por el tipo de repetición del evento por ello aunque lo definamos como 1 puede tener valores desde 1 hasta 51 (vea la referencia de temas para más detalles). Esta también es la razón de porque la prioridad del día actual es 100 en lugar de p.ej. 2.

El atributo *align* establece los números al centro de las celdas del calendario en malla.

Puedes haber notado que hemos usado el mismo color para los días normales y con evento lo cual no es muy inteligente ya que no es posible ver qué días tienen eventos. Corregiremos esto definiendo una nueva fuente y usándola en los eventos y en el día actual.

```
<elements>
  <font id="font.normal" facename="Arial" size="12" />
  <font id="font.bold" facename="Arial" weight="BOLD" size="12" />
</elements>

<calendar id="Calendar.calendar" x="0" y="0" w="200" h="200" layout="GRID" style="background-color:#f0f0f0">
  <days>
    <appearance layer="0" priority="0" element="font.normal" color="102,102,102" />
```

```

--
</days>
<today>
  <appearance layer="0" priority="100" element="font.bold" color="249,
</today>
<events>
  <appearance layer="0" priority="1" element="font.bold" color="102, :
</events>
</calendar>

```

Ahora podemos iniciar Rainlendar y revisar nuestro (incompleto) tema. Simplemente ve a Opciones->Temas y deberías ver MyFirstSkin en la lista. Si estás en modo Avanzado deberías cambiar al modo Simple. Selecciona el tema de la lista y deberías ver los detalles que agregamos al elemento <info> del tema. Haz click en Aceptar y deberías ver una ventana similar a esta:

```

1
2 3 4 5 6 7 8
9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30

```

Quite plain looking and the desktop wallpaper shows through so it's pretty hard to move the window or open the context menu.

A continuación debemos hacer algo con el fondo para que el texto sea más visible. Para hacer esto necesitamos primero crear un elemento para el fondo y entonces usarlo en la ventana. Para el fondo utilizaremos dos gradientes que irán en direcciones opuestas.

Los elementos:

```

<elements>
  <font id="font.normal" facename="Arial" size="12" />
  <font id="font.bold" facename="Arial" weight="BOLD" size="12" />
  <gradient id="gradient.background1" color1="201, 131, 59" color2="237,
  <gradient id="gradient.background2" color1="237, 170, 100" color2="20:
</elements>

```

... la ventana:

...y la ventana:

```
<window id="Calendar" w="200" h="200" threshold="48" default="1">
  <image id="Calendar.background1" element="gradient.background1" x="0"
  <image id="Calendar.background2" element="gradient.background2" x="3"
  <month id="Calendar.month" element="font.large" x="0" y="0" ALIGN="TOI
  <year id="Calendar.year" element="font.large" x="200" y="0" ALIGN="TOI
  <calendar id="Calendar.calendar" x="0" y="0" w="200" h="200" layout="(
    <days>
      <appearance layer="0" priority="0" element="font.normal" color="10
    </days>
    <today>
      <appearance layer="0" priority="100" element="font.bold" color="24
    </today>
    <events>
      <appearance layer="0" priority="1" element="font.bold" color="102,
    </events>
  </calendar>
</window>
```

Puedes notar que las imágenes del fondo deben ser las primeras entradas para la ventana. Los ítems son dibujados en el mismo orden en que son definidas por lo que si quieres superponer los ítems el que se encuentre detrás debe estar definido anteriormente.

Nuestra ventana de calendario se ve así:



Ahora debemos hacer algo de espacio para el mes y el año. También utilizaremos una fuente ligeramente más grande así que agregaremos la definición en la sección de los elementos.

```

<elements>
  <font id="font.normal" facename="Arial" size="12" />
  <font id="font.bold" facename="Arial" weight="BOLD" size="12" />
  <gradient id="gradient.background1" color1="201, 131, 59" color2="237,
  <gradient id="gradient.background2" color1="237, 170, 100" color2="201
  <font id="font.large" facename="Arial" weight="BOLD" size="14" />
</elements>

```

El mes y el año van en la parte superior de la ventana por lo que necesitamos mover el calendario un poco más abajo. También moveremos los límites del calendario ligeramente hacia adentro para que encaje mejor en el fondo.

```

<window id="Calendar" w="200" h="200" threshold="48" default="1">
  <image id="Calendar.background1" element="gradient.background1" x="0"
  <image id="Calendar.background2" element="gradient.background2" x="3"
  <month id="Calendar.month" element="font.large" x="5" y="5" align="TOP"
  <year id="Calendar.year" element="font.large" x="195" y="5" align="TOP"
  <calendar id="Calendar.calendar" x="5" y="35" w="190" h="160" layout="
    <days>
      <appearance layer="0" priority="0" element="font.normal" color="102
    </days>
    <today>
      <appearance layer="0" priority="100" element="font.bold" color="201
    </today>
    <events>
      <appearance layer="0" priority="1" element="font.bold" color="102,
    </events>
  </calendar>
</window>

```

Entonces, ahora tenemos el mes y el año en nuestro calendario. El mes se encuentra en el lado izquierdo del calendario en las coordenadas 5,5 y el año en el lado derecho en las coordenadas 195,5. La alineación del año debe establecerse a TOP-RIGHT para que el texto se desplace hacia la izquierda. De otro modo saldría de la ventana.



2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

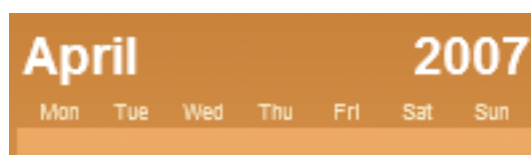
Lo último que necesitamos hacer a la ventana del calendario es agregarle los nombres de los días encima de los números de los días. Rainlendar los ubicará automáticamente así que no necesitamos preocuparnos por ello. Lo que necesitamos es crear una nueva fuente para los días de la semana y ajustar la posición del fondo en el calendario ligeramente (y definir los días de la semana por supuesto).

```

<elements>
  <font id="font.normal" facename="Arial" size="12" />
  <font id="font.bold" facename="Arial" weight="BOLD" size="12" />
  <font id="font.large" facename="Arial" weight="BOLD" size="14" />
  <font id="font.small" facename="Arial" size="6" />
  <gradient id="gradient.background1" color1="201, 131, 59" color2="237, 170, 100" />
  <gradient id="gradient.background2" color1="237, 170, 100" color2="201, 131, 59" />
</elements>

<window id="Calendar" w="200" h="200" threshold="48" default="1">
  <image id="Calendar.background1" element="gradient.background1" x="0" y="0" />
  <image id="Calendar.background2" element="gradient.background2" x="3" y="3" />
  <month id="Calendar.month" element="font.large" x="5" y="5" align="TOP" />
  <year id="Calendar.year" element="font.large" x="195" y="5" align="TOP" />
  <calendar id="Calendar.calendar" x="5" y="25" w="190" h="170" layout="CALENDAR"
    <weekdays abbreviate="1">
      <appearance element="font.small" color="249, 237, 204" align="CENTER" />
    </weekdays>
  <days>
    <appearance layer="0" priority="0" element="font.normal" color="102, 102, 102" />
  </days>
  <today>
    <appearance layer="0" priority="100" element="font.bold" color="249, 237, 204" />
  </today>
  <events>
    <appearance layer="0" priority="1" element="font.bold" color="102, 102, 102" />
  </events>
</calendar>
</window>

```



						1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						

El atributo *abbreviate* es utilizado para acortar los nombres de los días ya que los nombres completos no encajan en el espacio reservado.

Bien, eso es todo para la ventana del calendario. A continuación crearemos las listas de eventos y tareas y las otras ventanas.

[Anterior](#) - [Siguiete](#)

El skin.xml debería verse así:

```
<?xml version="1.0" encoding="UTF-8"?>
<skin version="1.0">

  <info>
    <version>1.0</version>
    <author>Rainy</author>
    <email>rainy@iki.fi</email>
    <homepage>http://www.rainlendar.net</homepage>
    <comment>The result of the skinning tutorial.</comment>
  </info>

  <elements>
    <font id="font.normal" facename="Arial" size="12" />
    <font id="font.bold" facename="Arial" weight="BOLD" size="12" />
    <font id="font.large" facename="Arial" weight="BOLD" size="14" />
    <font id="font.small" facename="Arial" size="6" />
    <gradient id="gradient.background1" color1="201, 131, 59" color2="201, 131, 59" />
    <gradient id="gradient.background2" color1="237, 170, 100" color2="237, 170, 100" />
  </elements>

  <window id="Calendar" w="200" h="200" threshold="48" default="1">
    <image id="Calendar.background1" element="gradient.background1" x="0" y="0" w="200" h="200" />
    <image id="Calendar.background2" element="gradient.background2" x="0" y="0" w="200" h="200" />
    <month id="Calendar.month" element="font.large" x="5" y="5" align="left" />
    <year id="Calendar.year" element="font.large" x="195" y="5" align="right" />
    <calendar id="Calendar.calendar" x="5" y="25" w="190" h="170" layout="vertical">
      <weekdays abbreviate="1">
```

```

    <appearance element="font.small" color="249, 237, 204" align="C"
</weekdays>
<days>
    <appearance layer="0" priority="0" element="font.normal" color='
</days>
<today>
    <appearance layer="0" priority="100" element="font.bold" color='
</today>
<events>
    <appearance layer="0" priority="1" element="font.bold" color="10
</events>
</calendar>
</window>

</skin>

```

Las Otras Ventanas

Hay cuatro otras ventanas que el tema puede necesitar: Lista de eventos, Lista de tareas, Alarma y Pistas. Podemos también definir el icono de la barra de tareas pero como eso requiere imágenes dejaremos el icono incorporado.

Para la lista de eventos crearemos una nueva ventana, estableceremos el fondo y definiremos el ítem de la lista.

```

<window id="Event List" w="200" threshold="48" default="1">
  <image id="Event List.background1" element="gradient.background1" ori
  <image id="Event List.background2" element="gradient.background2" ori
  <text id="Event List.header" element="font.large" x="5" y="5" align="C"
  <eventlist id="Event List.eventlist" origin1="TOP-LEFT" x1="10" y1="40"
    <header>
      <appearance element="font.bold" color="249, 237,204" align="TOP-LE
    </header>
    <item>
      <appearance element="font.normal" color="102, 35, 36" align="TOP-LE
    </item>
    <separator id="Event List.separator" padding="0, 10, 0, 0" />
  </eventlist>
</window>

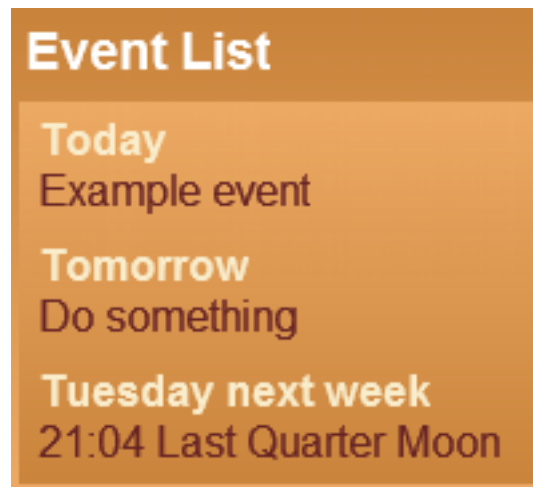
```

Un par de hechos que deberías notar acerca de la ventana de lista de eventos: la ventana no define ninguna altura y las imágenes de fondo y la lista de eventos no definen un tamaño absoluto. Esto es porque la ventana automáticamente ajustará su altura por los contenidos de la lista. Las imágenes de fondo están definidas para que siempre llenen la ventana completa desde arriba a la izquierda hasta abajo a la derecha. El *minsizefromimage* debe estar establecido a "0" para las imágenes. Esto es porque el

gradiente es en realidad una imagen de 1x256 de tamaño así que a menos que le digamos a Rainlendar que está bien encoger la imagen, establecerá el tamaño mínimo de la ventana a 256 lo que no se verá tan bien.

El elemento text simplemente dibuja la cabecera en la ventana. El texto también es traducido automáticamente si es encontrado en las cadenas traducidas.

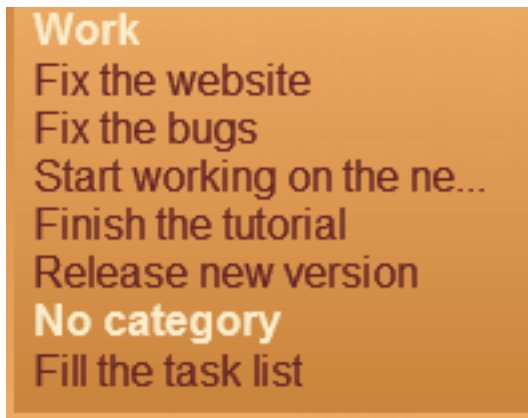
La verdadera posición de la lista de eventos es definida de la misma forma que el fondo pero con algunos rellenos en los bordes. La lista de eventos contiene la cabecera y los ítems que definen la apariencia de los textos. El separador es utilizado para añadir algunos espacios entre los ítems (puede también dibujar p.ej. una imagen entre los ítems).



La lista de tareas es muy similar a la lista de eventos. En realidad la única diferencia es el atributo *id* y por ello crearemos un `<todolist>` en lugar de un `<eventlist>`.

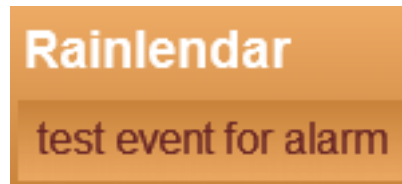
```
<window id="Todo List" w="200" threshold="48" default="1">
  <image id="Todo List.background1" element="gradient.background1" orig:
  <image id="Todo List.background2" element="gradient.background2" orig:
  <text id="Todo List.header" element="font.large" x="5" y="5" align="T
  <todolist id="Todo List.todolist" origin1="TOP-LEFT" x1="10" y1="40" <
    <header>
      <appearance element="font.bold" color="249, 237, 204" align="TOP-1
    </header>
    <item>
      <appearance element="font.normal" color="102, 35, 36" align="TOP-1
    </item>
    <separator id="Todo List.separator" padding="0, 10, 0, 0" />
  </todolist>
</window>
```





La ventana alarma no es muy diferente del resto.

```
<alarm id="Alarm" threshold="48" textmargins="10, 40, 10, 10">
  <image id="Alarm.background1" element="gradient.background1" origin1='
  <image id="Alarm.background2" element="gradient.background2" origin1='
  <text id="Alarm.header" element="font.large" x="5" y="5" align="TOP-LI
  <item id="Alarm.item">
    <appearance element="font.normal" color="102, 35, 36" align="TOP-LEI
  </item>
  <button id="Alarm.button.dismissall" origin1="TOP-LEFT" x1="0" y1="0"
</alarm>
```



La alarma no es `<window>` pero utiliza una etiqueta `<alarm>` en su lugar. Nunca se define el tamaño de la alarma porque es calculada automáticamente. El *textmargins* define el área alrededor de la ventana de alarma donde son agregados los márgenes para el texto. La ventana de alarma puede contener los mismos ítems que una ventana normal pero en adición define los elementos `<item>` de la lista que contiene la apariencia para los ítems de la alarma.

En la alarma también definidos un botón transparente (es transparente porque no definimos un elemento gráfico para él) que cubre toda la ventana. Presionando el botón descartaremos todas las alarmas que contiene la ventana. *Global_DismissAll()* es el nombre del script que es ejecutado cuando el botón es presionado. Puedes verificar la carpeta de scripts de Rainlendar para los scripts adicionales que pueden ser ejecutados. En la ventana de alarma puedes tener además un botón que duerma las alarmas.

La última ventna que tenemos que definir es la de pistas.

```
<tooltip id="Tooltip" threshold="48" textmargins="10, 10, 10, 10" offset
  <image id="Tooltip.background1" element="gradient.background1" origin1='
```

```

<image id="Tooltip.background1" element="gradient.background1" origin:
<image id="Tooltip.background2" element="gradient.background2" origin:
<header>
  <appearance element="font.bold" color="102, 35, 36" align="TOP-LEFT"
</header>
<item>
  <appearance element="font.normal" color="102, 35, 36" align="TOP-LEI
</item>
<separator id="Tooltip.separator" padding="0, 10, 0, 0" />
</tooltip>

```



Eso es prácticamente igual a la ventana de alarma. Las pistas tienen un elemento especial como la alarma y también define los márgenes del texto. *offset* define que tan lejos del puntero del ratón es abierta la ventana. Esto es útil si la pista tiene p.ej. una flecha que debería estar debajo del ratón. En nuestro ejemplo la pista es simplemente un rectángulo por lo que no definiremos desplazamiento alguno.

El resto es casi una copia de la ventana de alarmas con la excepción de que las pistas no poseen título (aunque podría...).

[Prev](#) - [Next](#)

El skin.xml finalizado debería ser ahora como lo siguiente:

```

<?xml version="1.0" encoding="UTF-8"?>
<skin version="1.0">
  <info>
    <version>1.0</version>
    <author>Rainy</author>
    <email>rainy@iki.fi</email>
    <homepage>http://www.rainlendar.net</homepage>
    <comment>The result of the skinning tutorial.</comment>
  </info>

```

```

<elements>
  <font id="font.normal" facename="Arial" size="12" />
  <font id="font.bold" facename="Arial" weight="BOLD" size="12" />
  <font id="font.large" facename="Arial" weight="BOLD" size="14" />
  <font id="font.small" facename="Arial" size="6" />
  <gradient id="gradient.background1" color1="201, 131, 59" color2="2:
  <gradient id="gradient.background2" color1="237, 170, 100" color2=":
</elements>

<window id="Calendar" w="200" h="200" threshold="48" default="1">
  <image id="Calendar.background1" element="gradient.background1" x="(
  <image id="Calendar.background2" element="gradient.background2" x=":
  <month id="Calendar.month" element="font.large" x="5" y="5" align=":
  <year id="Calendar.year" element="font.large" x="195" y="5" align=":
  <calendar id="Calendar.calendar" x="5" y="25" w="190" h="170" layout
    <weekdays abbreviate="1">
      <appearance element="font.small" color="249, 237, 204" align="CI
    </weekdays>
    <days>
      <appearance layer="0" priority="0" element="font.normal" color='
    </days>
    <today>
      <appearance layer="0" priority="100" element="font.bold" color='
    </today>
    <events>
      <appearance layer="0" priority="1" element="font.bold" color="10
    </events>
  </calendar>
</window>

<window id="Event List" w="200" threshold="48" default="1">
  <image id="Event List.background1" element="gradient.background1" o:
  <image id="Event List.background2" element="gradient.background2" o:
  <text id="Event List.header" element="font.large" x="5" y="5" align=:
  <eventlist id="Event List.eventlist" origin1="TOP-LEFT" x1="10" y1='
    <header>
      <appearance element="font.bold" color="249, 237,204" align="TOP-
    </header>
    <item>
      <appearance element="font.normal" color="102, 35, 36" align="TOI
    </item>
    <separator id="Event List.separator" padding="0, 10, 0, 0" />
  </eventlist>
</window>

<window id="Todo List" w="200" threshold="48" default="1">
  <image id="Todo List.background1" element="gradient.background1" or:
  <image id="Todo List.background2" element="gradient.background2" or:
  <text id="Todo List.header" element="font.large" x="5" y="5" align='

```

```

<todolist id="Todo List.todolist" origin1="TOP-LEFT" x1="10" y1="40"
  <header>
    <appearance element="font.bold" color="249, 237, 204" align="TOP-LEFT" />
  </header>
  <item>
    <appearance element="font.normal" color="102, 35, 36" align="TOP-LEFT" />
  </item>
  <separator id="Todo List.separator" padding="0, 10, 0, 0" />
</todolist>
</window>

<tooltip id="Tooltip" threshold="48" textmargins="10, 10, 10, 10" offset="0, 0" />
  <image id="Tooltip.background1" element="gradient.background1" origin="TOP-LEFT" />
  <image id="Tooltip.background2" element="gradient.background2" origin="TOP-LEFT" />
  <header>
    <appearance element="font.bold" color="102, 35, 36" align="TOP-LEFT" />
  </header>
  <item>
    <appearance element="font.normal" color="102, 35, 36" align="TOP-LEFT" />
  </item>
  <separator id="Tooltip.separator" padding="0, 10, 0, 0" />
</tooltip>

<alarm id="Alarm" threshold="48" textmargins="10, 40, 10, 10">
  <image id="Alarm.background1" element="gradient.background1" origin="TOP-LEFT" />
  <image id="Alarm.background2" element="gradient.background2" origin="TOP-LEFT" />
  <text id="Alarm.header" element="font.large" x="5" y="5" align="TOP-LEFT" />
  <item id="Alarm.item">
    <appearance element="font.normal" color="102, 35, 36" align="TOP-LEFT" />
  </item>
  <button id="Alarm.button.dismissall" origin1="TOP-LEFT" x1="0" y1="0" />
</alarm>

</skin>

```

Finalizando Las Cosas

Lo último que tenemos que hacer es empaquetar el tema dentro de un archivo r2skin. El archivo está comprimido de forma estándar así que podemos usar por ejemplo la funcionalidad archivadora incorporada en Windows. Cuando creamos el paquete debemos asegurarnos que el skin.xml está en la carpeta raíz del archivo. Si hay una subcarpeta extra en el archivo Rainlendar no será capaz de encontrar el skin.xml de él y el tema no estará visible en la lista de temas.

En Windows puedes hacer click derecho en el skin.xml y seleccionar "Enviar a" -> "Carpeta Comprimida (zipeada)" (si tú tema contiene múltiples archivos,

selecciónalos todos). Esto creará un nuevo archivo llamado skin.zip dentro de la carpeta *Nahka*. Renómbralo a Nahka.r2skin y muévelo a la carpeta superior. También deberías eliminar o renombrar la carpeta *Nahka* anterior. El Nahka.r2skin está ahora listo para ser distribuido a otros usuarios.

El tema Nahka está todo en un solo archivo pero si estás implementando un tema complejo sería útil definir las diferentes ventana en archivos xml separados para que el archivo no crezca demasiado (a Rainlendar no le preocupa pero es probablemente más fácil encontrar cosas si los archivos son más pequeños). Puedes verificar el tema Shadow4 como ejemplo de cómo hacerlo.

[Anterior](#) - Siguiente