Archived:Example CV in WidSets

Archived: This article is **archived** because it is not considered relevant for third-party developers creating commercial solutions today. If you think this article is still relevant, let us know by adding the template {{ReviewForRemovalFromArchive|user=~~~~|*write your reason here*}}.

The article is believed to be still valid for the original topic scope.

This is a very simple example widget to show how easy it is to create customized WidSets widgets. To create your own customized CV widget, just fill in your own information to the widget code and add some new content components.

Read the comments in the code to understand it better.

cv.he

```
class cv
{
  const int CMD_BACK = 1;                                                    //
Constant integer for softkey action.

  MenuItem BACK = new MenuItem(CMD_BACK, "Back");                            //
MenuItem that will be bind as SOFTKEY_BACK.

  void startWidget() {                                                       //
Function that is called when client starts. Creates the minimized view to
                                                                             //
dashboard.

    setMinimizedView(createMinimizedView("minimizedView", null));            //
MinimizedView is created according to minimizedView layout defined in widget.xml.
  }

  Shell openWidget() {                                                       //
Function that is called when widget is opened. Returns the Shell that is the
                                                                             //
actual "screen" that opens.

    return createStartShell();                                               //
Function that returns Shell is called and returned straight to client core for
                                                                             //
widget opening.

  }

  Shell createStartShell() {                                                 //
Function that creates the content Shell and returns it to openWidget function.

    Flow flow = new Flow(getStyle("background"));                            // Flow
is a component container where all the content components are placed.
                                                                             // Style
"background" is used, which sets the background white.
```


(C) Copyright Nokia 2014. All rights reserved.

```
    flow.setPreferredSize(-100,-100);                                   // Flow
is set to fill the whole screen so that the whole screen is white even
                                                                        // if
the content does not fill the screen.

    Text text = new Text(getStyle("largeText"), "Test Users CV");       // Text
component with content title is created
    text.setPreferredWidth(-100);                                       // width
is set to 100% so when the horizontal alignment is set to center, the text
                                                                        //
aligns in the center of the screen.

    text.setFlags(VISIBLE|LINEFEED);                                    // Title
is given flags VISIBLE and LINEFEED, resulting that it is shown
                                                                        // and
the next component in the container comes under it.

    flow.add(text);                                                     // Title
Text component is added to the flow container.

    Picture avatar = new Picture(getStyle("center"), getImage("avatar.png"));   // New
Picture component is created and avatar.png image is set in it.
                                                                        //
Avatar.png is a resource file.

    avatar.setFlags(VISIBLE|LINEFEED);                                  //
Avatar Picture gets the same flags as title.

    flow.add(avatar);                                                   //
Avatar is added to flow.

                                                                        // Add
content components here. For example, education, former jobs, etc.
                                                                        // Use
Text, Label, Picture, etc. components. See the API documentation and
                                                                        //
http://dev.widsets.com/wiki for more help.

    return new Shell(new Scrollable(getStyle("background"), flow));     // Flow
is placed inside the Scrollable component in case the content exceeds
                                                                        // the
size of the screen. Scrollable is places in Shell that is returned.
  }

  MenuItem getSoftKey(Shell shell, Component focused, int key) {        // This
function sets bindings for the softkeys.

    if (key == SOFTKEY_BACK) {                                          //
Binding for the SOFTKEY_BACK is mandatory to enable the closing of the widget.

      return BACK;                                                      // BACK
MenuItem is binded for the SOFTKEY_BACK.
    } else {
      return null;                                                      // Null
is returned for SOFTKEY_MIDDLE and SOFTKEY_OK.
    }
  }
```


(C) Copyright Nokia 2014. All rights reserved.

```
  void actionPerformed(Shell shell, Component source, int action) {             // This
function catches the user actions.

    switch(action) {                                                             //
Switch statement is useful if there are many different actions happening in the
                                                                                 //
widget.

      case CMD_BACK:                                                             // In
the case of CMD_BACK action (when the SOFTKEY_BACK is pressed),
        popShell(shell);                                                         // the
Shell disappeares from the screen and dashboard becomes visible again.
        break;
    }
  }
}
```

widget.xml

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

<widget spec_version="2.0">
<!-- go to http://dev.widsets.com/wiki/Widget_Configuration_2.0 for help -->

  <info>
    <name>widget name</name>
    <version>0.1</version>
    <author>creators name</author>
    <shortdescription>short description</shortdescription>
    <longdescription>long description</longdescription>
    <tags>CV cool</tags>
  </info>

  <resources>
    <img src="avatar.png"/>
    <code src="cv.he"/>
    <stylesheet>
      minimized {
        align: hcenter vcenter;
        background: solid white;
        border: 1 1 1 1;
        border-type: rectangle black;
      }
      smallText {
        align: hcenter vcenter;
        color: black;
        font: small;
      }
      largeText : smallText {
        font: large bold;
      }
      .background {
        align: hcenter top;
        background: solid white;
      }
```


(C) Copyright Nokia 2014. All rights reserved.

```
    </stylesheet>
  </resources>

  <layout minimizedheight="40sp">
    <view id="minimizedView">
      <decorate class="minimized"/>
      <text class="largeText">My CV</text>
    </view>
  </layout>
</widget>
```


(C) Copyright Nokia 2014. All rights reserved.