How to monitor Device Orientation Change in WRT Widget

Overview

Note: Support for S60 Platform Services through **Platform Services APIs** was introduced in **WRT 1.1**. The Platform Services 1.0 APIs were formerly known as **Service APIs**. Widgets created using WRT 1.1 Platform Services APIs do not work with WRT 1.0.

- This article and code snippet below explains how one can monitor changes in **Device Orientation** in a Widget using Platform Services Sensors (Orientation) API. There are two versions of this particular API, one for Platform Services 1.0 and one for Platform Services 2.0. Both versions are demonstrated in this snippet.

- Types of device orientation

Using Platform Services 1.0

```
<html>
    <head>

        <script language="javascript">

        var so = null;
        var channel = "Orientation";
        var tid = "";

        // Callback method called whenever change in Device Orientation is detected.
        function onOrientationChange( tid, eventCode , result ){
          if(result.ErrorCode != 0){
            alert("Error :"  + result.ErrorMessage);
            return;
          }

          alert( "Device Orientation : " + result.ReturnValue.DeviceOrientation );
        };

        // On exit lets stop the sensor.
        window.widget.onexit = function() {

          if(tid ) {
            var params = {
              TransactionID : tid
            };
           var result = so.ISensor.Cancel(params);
           if(result.ErrorCode != 0 ){
             alert("Error stopping sensor :" + result.ErrorMessage);
            }
         }
        };


        try{
          // Lets first create sensor object.
          so = device.getServiceObject("Service.Sensor","ISensor");
         var params = {
           SearchCriterion : channel
         };

         var result = so.ISensor.FindSensorChannel( params );

         if(result.ErrorCode == 0 && result.ReturnValue.length > 0 ){

           // Lets start getting Device Orientation Changes.
```


(C) Copyright Nokia 2013. All rights reserved.

```
            var params  = {
        ListeningType: "ChannelData",
              ChannelInfoMap: result.ReturnValue[0]
          };

         var result2 = so.ISensor.RegisterForNotification ( params , onOrientationChange);
            // Lets also save the transaction Id to cancel on going RegisterForNotification on exit.
            tid = result2.TransactionID;

         }else{
            alert("Error getting channel :" + result.ErrorMessage);
          }
        }catch(e){

          alert("Error : " + e);
        };


        </script>
    </head>
    <body>Device Orientation Test Widget <br> using Platform Services 1.0</body>
</html>
```

Sample Widget

Here is a sample widget, uses Platform Services 1.0 Sensors API

Using Platform Services 2.0

- Before we start with the Platform Services 2.0, there are two ways of including Platform Services 2.0 support to the widget:

```
 - Installing platformservices_v2_0_beta.SIS, will add the JavaScript library to a common location
   (c:\system\widgetlibs\platformservices.js) that can be accessed by any widget.
```

```
 - Including the JavaScript library platformservices.js directly in a widget.
```

- We will use the second option for simplicity, and ease of testing the widget.
- In the code below we try to monitor the changes in the device Orientation and display it on screen, for that we will use **Orientation Sensors** api.

```
<html>
    <head>
        <script type="text/javascript" src="platformservices.js" charset= "utf-8" />

        <script language="javascript">

        var so = null;
        var channel = "Orientation";

        // Callback method called whenever change in Device Orientation is detected.
        function onOrientationChange( transId, errCode, orientation ){

          if( errCode == 0 ){
            alert("Device Orientation : " + orientation.deviceOrientation);
          }
        };

        // On exit lets stop sensor.
        window.widget.onexit = function() {

            so.stopChannel(channel);
        };


        try{
          // Lets first create geolocation object.
          so = com.nokia.device.load("", "com.nokia.device.sensors");
          // Lets get the Orientation channel.
          var channels = so.getChannels();
          var len = channels.length;
          var sensorExists = false;
          for( var i = 0; i < len; i++){
            if( channels[i] === channel ){
              sensorExists = true;
              break;
            }
          }

          if( !sensorExists ){
               alert("Device Orientation sensor not available !");
               return;
          }

          // Lets start getting Device Orientation Changes.
          var tid = so.startChannel(onOrientationChange, channel);

        }catch(e){

          alert("Error : " + e);
        };


        </script>
    </head>
    <body>Device Orientation Test Widget <br>using platform services 2.0</body>
</html>
```


(C) Copyright Nokia 2013. All rights reserved.

Sample Widget

Here is a sample widget, uses Platform Services 2.0 Sensors API

How to Test

- Launch the widget, it will start monitoring device orientation and will display the orientation if changes are detected in the device orientation.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.