Archived:Listening for rotation sensor data events on Symbian

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*}}.

Overview

This code snippet demonstrates how to listen for data events produced by the rotation sensor of the device. In practice, the control implements the `MRRSensorDataListener` interface which enables it to listen to the data events. Whenever a data event occurs, the `MRRSensorDataListener::HandleDataEventL()` function is called.

Note: In order to use the code, you need to install the sensor plug-in for your SDK.

This snippet can be self-signed.

MMP file

The following libraries are required:

```
LIBRARY RRSensorApi.lib
```

Header file

```
#include <RRSensorApi.h>
```

```
// Rotation sensor UID
const TInt KRotationSensorUID = 0x303E;
```

```
class CMyControl : public CCoeControl, public MRRSensorDataListener
    {
    // ...

    private:  // Functions from base classes
        /**
         * From MRRSensorDataListener.
         * Callback function for receiving sensor data events.
         *
         * @param aSensor identifies the sensor that created the event.
         * @param aEvent contains data about created event.
         */
        void HandleDataEventL(TRRSensorInfo aSensor, TRRSensorEvent aEvent);

    private:
        /**
         * Initializes and registers the rotation sensor listener.
         */
        void RegisterSensors();

        /**
         * Unregisters the rotation sensor listener.
         */
        void UnregisterSensors();

    private:  // Data
        CRRSensorApi* iRotationSensor;
    }
```

Source file

```
#include <RRSensorApi.h>
```

```
void CMyControl::ConstructL(const TRect& aRect)
    {
    // ...

    // Initialize and register the rotation sensor listener
    RegisterSensors();

    // ...
    }
```

```
CMyControl::~CMyControl()
    {
    // ...
```


(C) Copyright Nokia 2013. All rights reserved.

```
    // Unregister the rotation sensor listener
    UnregisterSensors();

    // ...
    }
```

```
/**
 * Initializes and registers the rotation sensor listener.
 */
void CMyControl::RegisterSensors()
    {
    RArray<TRRSensorInfo> sensorList;
    CleanupClosePushL(sensorList);

    // Retrieve list of available sensors
    CRRSensorApi::FindSensorsL(sensorList);

    // Get number of sensors available
    TInt sensorCount = sensorList.Count();

    for (TInt i = 0; i < sensorCount; i++)
        {
        // We are interested only in the rotation sensor now
        if (sensorList[i].iSensorId == KRotationSensorUID)
            {
            iRotationSensor = CRRSensorApi::NewL(sensorList[i]);

            // Register this control as rotation data listener
            iRotationSensor->AddDataListener(this);

            break;
            }
        }

    CleanupStack::PopAndDestroy();  // sensorList
    }
```

```
/**
 * Unregisters the rotation sensor listener.
 */
void CMyControl::UnregisterSensors()
    {
    // Unregister rotation data listener
    iRotationSensor->RemoveDataListener();
    delete iRotationSensor;
    iRotationSensor = NULL;
    }
```

```
/**
 * Gets called whenever a data event occurs.
 */
void CMyControl::HandleDataEventL(TRRSensorInfo aSensor, TRRSensorEvent aEvent)
    {
    // A data event occurred. Handle it.
    // ...
    }
```

Postconditions

Whenever a data event occurs, the `CMyControl::HandleDataEventL()` function is called.

See also

- Archived:Listening for accelerometer sensor data events on Symbian
- Archived:Displaying sensor data on Symbian

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.