

Giving tactile feedback in touch UI apps using Symbian C++

 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.

Overview

Tactile feedback provides the user a physical pulse (vibration) when touch screen interaction has been successful.

Tactile feedback provides an interface to add, modify, and remove feedback areas in the registry. There is also an option to trigger direct feedback by bypassing the registry.

MTouchFeedback is used for acquiring a pointer to the touch feedback instance.

This snippet can be self-signed.

MMP file

The following capabilities and libraries are required:

```
CAPABILITY None
```

```
LIBRARY touchfeedback.lib
```

Header file

Include two headers

```
#include <touchfeedback.h>
#include <touchlogicalfeedback.h>

class CImageConverterContainer : public CCoeControl
{
    ...
private:
    MTouchFeedback*      iTouchFeedBack;
    ...
}
```

Source file

```
// Create an instance of touch feedback
void CImageConverterContainer::ConstructL(const TRect& /*aRect*/)
{
    ...
    iTouchFeedBack = MTouchFeedback::Instance();
}
```

```
iTouchFeedBack->SetFeedbackEnabledForThisApp(ETrue);
...
}

// Give some vibra to the user when a pointer down event occurs
void CImageConverterContainer::HandlePointerEventL(const TPointerEvent& aPointerEvent)
{
    ...
    if (aPointerEvent.iType == TPointerEvent::EButton1Down)
    {
        // Give feedback to user (vibration)
        iTouchFeedBack->InstantFeedback(ETouchFeedbackBasic);
    }
    ...
}
```

Parameters into MTouchFeedback::InstantFeedback().

```
* ETouchFeedbackNone      - Use for disabling feedback for some areas of the
*                          window when using area registry.
*
* ETouchFeedbackBasic     - Use as default feedback for pen down events.
*                          For example, pressing a button or tab.
*
* ETouchFeedbackSensitive - Sensitive feedback situations
*                          where the triggering action is not very
*                          important (e.g. change focus in list), or when
*                          there can be a large amount of feedback in
*                          a short time (e.g. text selection which gives
*                          feedback on every new selected character).
*                          Also used for scrolling and dragging.
*
```

Postconditions

The user gets vibration feedback when using the stylus / finger down (tapping on the screen).

See also

[S60 Platform: Image Converter Example](#)

