# How to use SoftNotification API in Java ME

## Overview

27 Sep 2009

Java Runtime 1.3 for S60 introduced a new UI feature, Soft Notification API. It is an extension of Nokia UI API 1.2 and it is commonly available in S60 5th Edition devices (and newer). Soft notifications inform the user of events that have occurred in the user's absence, or while the user was busy with some application. They are displayed as pop-ups in device home screen (idle view). It is possible to set the notification text and the icon image for a notification. Several image formats are supported, for example png and svg image formats can be used.

## Usage

This article introduces EggTimer MIDlet, which demonstrates the usage of soft notifications. The `TimerNotification` class contains the most relevant pieces of code. In addition to the `TimerNotification` class there is the main `MIDlet` class `EggTimer.java` and `KeypadCanvas` class, which is used for creating the UI for entering the timer time. The actual timer functionality is handled by using standard `java.util.TimerTask` and `java.util.Timer` classes.

Soft notifications can be displayed for the user in two different appearances, ungrouped and grouped. Ungrouped soft notifications contain one piece of information each. Ungrouped appearance is used when there's only one active soft notification or the topmost soft notification does not support grouping. Note also, that soft notifications are persistent and restarting the phone does not delete created soft notifications.

Many different items of information can be combined into one grouped soft notification where the items are displayed as a list. The user can pick up one of the items at a time and react to it. If user discards the grouped soft notification, then all the shown items are discarded and are not to be shown again.

The Soft Notification API consists of two new classes and one interface in Nokia UI API's com.nokia.mid.ui package:

- SoftNotificationListener interface
- SoftNotification class
- SoftNotificationException class

This API is an optional feature in Nokia UI API, and depending on the device capabilities there may be implementations that do not support soft notifications. The System.getProperty("com.nokia.mid.ui.softnotification") will return "true" if this feature is available in the device. The following code shows, how a soft notification is created and how its features are set. Finally it is shown to user by using the SoftNotification.post() method.

```
private void showTimerNotification() throws SoftNotificationException {
 softNotification = SoftNotification.newInstance();
 softNotification.setListener(this);
 softNotification.setText("Note! Timer time passed: " + time + " minutes.", "");
 softNotification.setSoftkeyLabels("Ok", "Exit");
 softNotification.setImage(readResourceImage("/egg.png"));
 softNotification.post();
}
```

The user can respond to the soft notification by using the softkeys. The softkey 1 (left soft key) is used for activating a function, for example opening a message that has arrived. Application that created the soft notification will be brought to the foreground when the soft notification is accepted by the user. The softkey 2 (right soft key) is used to discard the notification without taking any further action. When softkey 1 is selected, SoftNotificationListener.notificationSelected() method is called. In case of selecting softkey 2 is selected, SoftNotificationListener.notificationDismissed() method is called.

Below a code snippet shows the implementations of these two methods. In both cases the alarm sound is stopped. If softkey 1 is selected, the EggTimer MIDlet is automatically brought to the foreground. This happens, even if the MIDlet has been closed during the timer time. In case of selecting softkey 2 MIDlet is closed.


(C) Copyright Nokia 2014. All rights reserved.

```
public void notificationSelected(SoftNotification notification) {
 try {
  midlet.player.stop();
  midlet.player.close();
  midlet.player = null;
  time = 0;
 } catch (MediaException me) {
  midlet.showError("MediaException", me.getMessage());
 }
}

public void notificationDismissed(SoftNotification notification) {
 try {
  midlet.player.stop();
  midlet.player.close();
  midlet.player = null;
  time = 0;
 } catch (MediaException me) {
  midlet.showError("MediaException", me.getMessage());
 }
 midlet.notifyDestroyed();
}
```

## Recognizing keyPresses by using scancodes

In addition to the using touch screen it is also possible to enter timer time by using numeric keys in EggTimer MIDlet. For example in N97 number keys are in the upper row of the keyboard. Normally key events for number can be caught by using the keys together with the FN key. For example, pressing FN + Q will give the key code of "1" key. Another option is to use scan codes, as shown below:

```
protected void keyPressed(int keyCode) {
 int selected = 0;  // the selected number key
 keyScanCode = 0; // the scan code of the selected key
 try {
  keyScanCode = Integer.parseInt(System.getProperty("com.nokia.key.scancode"));
 }
 catch (NumberFormatException nfe) {
  midlet.showError("NumberFormatException", nfe.getMessage());
 }
 if (keyScanCode == 81) selected = 1;
 else if (keyScanCode == 87) selected = 2;
 else if (keyScanCode == 69) selected = 3;
 else if (keyScanCode == 82) selected = 4;
 else if (keyScanCode == 84) selected = 5;
 else if (keyScanCode == 89) selected = 6;
 else if (keyScanCode == 85) selected = 7;
 else if (keyScanCode == 73) selected = 8;
 else if (keyScanCode == 79) selected = 9;
 else if (keyScanCode == 80) selected = 0;
}
```

## Reading image data to byte array

As shown in the code above, it is possible to set the icon image for the soft notification. Images will be scaled to fit in the soft


(C) Copyright Nokia 2014. All rights reserved.

notification pop-up as the image area of the notification is relatively small. For saving resources in scaling, the size of the image should be kept small (preferably no larger than 300x300 px). Image data is given as byte array to the setImage(byte[] image) method. If the image is included in the MIDlet jar file, the image data can be read to a byte array by using the following method:

```
private byte[] readResourceImage(String resource) {
 byte[] byteData = null; // resource data
 try {
  InputStream is = getClass().getResourceAsStream(resource);
  if(is == null){ }
  else {
   DataInputStream dataStream = new DataInputStream(is);
   int byteRead = 0;
   ByteArrayOutputStream baos = new ByteArrayOutputStream();
   while (byteRead != -1){
    byteRead = dataStream.read();
    baos.write(byteRead);
   }
   byteData = baos.toByteArray();
   baos.close();
   dataStream.close();
  }
 }
 catch (IOException ioe) {
  midlet.showError("IOException", ioe.getMessage());
 }
 return byteData;
}
```

## Using Timer and TimerTask classes for alerting

The timer time is given in minutes. The given time is used in TimerNotification.startTimer() method, where the Timer and EggTimerTask instances are created. The timer.schedule method is used for scheduling the task to be executed after the set delay.

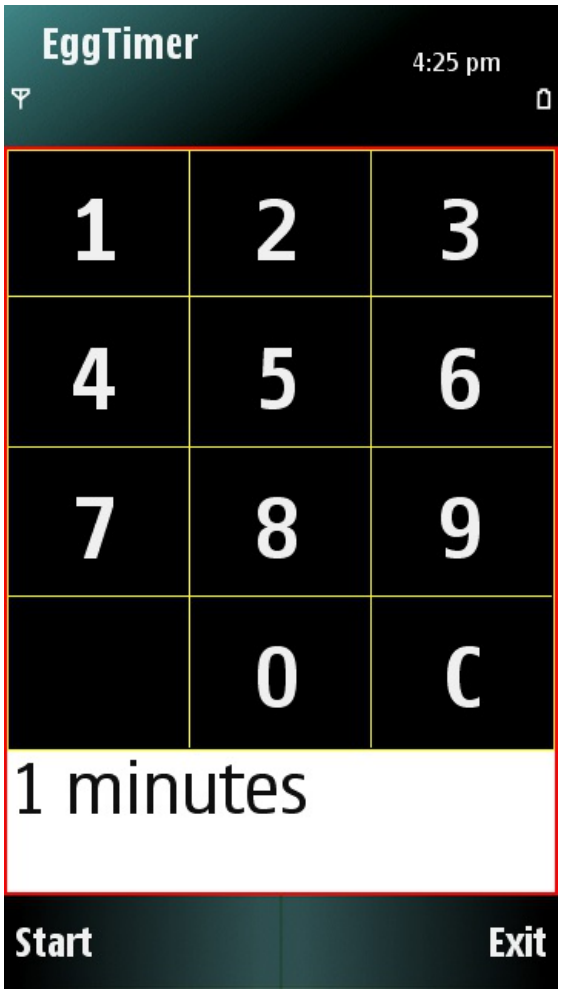
```
protected void startTimer() {
 timer = new Timer();
 task = new EggTimerTask();
 long millisecs = time*60*1000;
 timer.schedule(task, millisecs);
}
```

The scheduled task is in the EggTimerTask.run() method. The timer notification is shown on the screen and tone is played repeatedly, if the device idle screen is shown, or otherwise only a tone is played.

```
public final void run() {
 try {
  showTimerNotification();
 } catch (SoftNotificationException sne) {
  midlet.showError("SoftNotificationException", sne.getMessage());
 }
 playTimerTone();
}
```

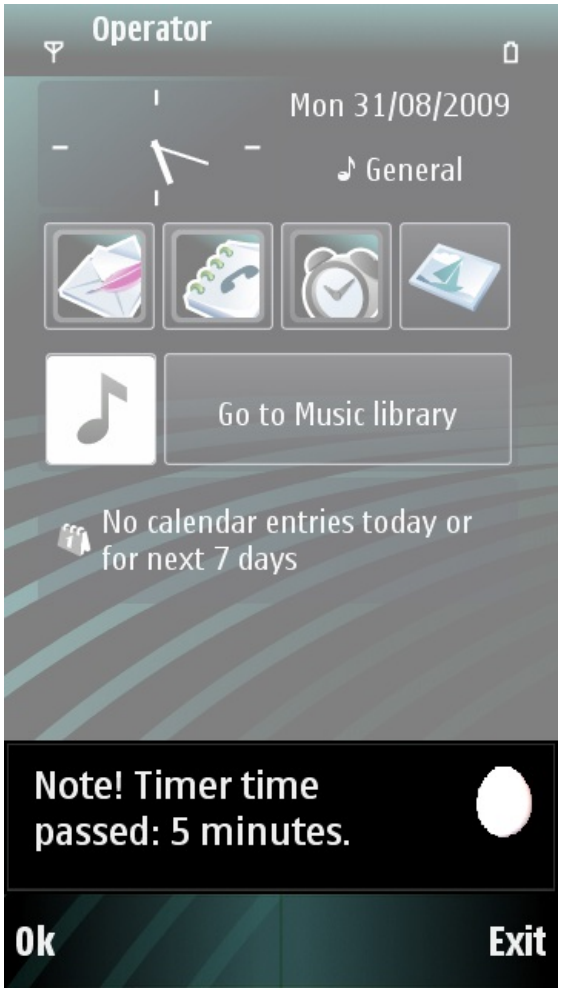

(C) Copyright Nokia 2014. All rights reserved.

The image below shows the EggTimer MIDlet running in Nokia N97 SDK.




The image below shows the soft notification shown by the EggTimer MIDlet running in Nokia N97 SDK.


(C) Copyright Nokia 2014. All rights reserved.




## Example application

- EggTimer.zip containing sources, EggTimer.jad and EggTimer.jar files

## See also

- SoftNotification class javadoc in Nokia Developer Java Developer's Library
- Nokia UI API's UI extensions


(C) Copyright Nokia 2014. All rights reserved.