



| | | | |
|-----------------|---|--------------------------|---------------|
| ID | ... | Creation date | 10 April 2009 |
| Platform | S60 3rd Edition FP1, S60 3rd Edition FP2, S60 5th Edition | Tested on devices | S60 Emulator |
| Category | Qt for Symbian | Subcategory | |

Keywords (APIs, classes, methods, functions): Q

Preconditions

- Download and Install latest version [Qt for Symbian - Installation packages](#) which has links on how to install the latest version

Dynamic Layout handling with QWidget

To be able to catch the layout change event, you first need to install event filter, which will give you then the events when layout changes happen. This can be done for example in the main.cpp, in where you can use `installEventFilter` function of the `QApplication` to install a filter, for example like this:

```
QApplication a(argc, argv);
QtEggAlarm w;
a.installEventFilter(&w);
```

The event filter class then needs to implement `eventFilter` function which will then be called when events happen. For layout changes a `QEvent::Resize` event will be generated, thus you could catch the event in the event filter as follows:

```
bool QtEggAlarm::eventFilter(QObject* /*receiver*/, QEvent* event)
{
    if(event->type() == QEvent::Resize)
    {
        SetScreenSizeAndPosition
    }
    return true;
}

return false;
}
```

Note that if the return value is false, then the event will be given to other filters, and if it is true, it will not be forwarded to other components.

Dynamic_Layout_handling_with_QWidget_in_Qt_for_Symbian

Then If you want to Widget to fill the whole client area in S60 device, the SetScreenSizeAndPosition function could be implemented as follows:

```
void QtEggAlarm::SetScreenSizeAndPosition(void)
{
    setGeometry(QApplication::desktop()->availableGeometry());
}
```

Screen shots



Example project

[QtEggAlarm.zip](#)