Archived:Send SMS using Qt Mobility

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*}}.

This example is outdated and may not compile with new sdks; see this for new examples Qt Mobility API Basic Examples Part 1

The following code shows how to send an SMS using Qt Mobility.

Project configuration file (.Pro file)

- Add the Qt Mobility project configuration option in the .Pro file as shown below

```
CONFIG += mobility
MOBILITY += messaging
```

The use of Messaging requires the ReadUserData and WriteUserData capabilities from the **user granted** category. It also requires the ReadDeviceData and WriteDeviceData capabilities from the **open signed online** category. Read more about Capabilities.

```
symbian {
TARGET.CAPABILITY = ReadUserData \
    WriteUserData \
    ReadDeviceData \
    WriteDeviceData
}
```

Headers

```
#include <qtmessaging.h>

private slots:
    void messageStateChanged(QMessageServiceAction::State s);
private:
    QMessageServiceAction m_MessageServiceAction;
    QMessageId m_sendId;
```

Source file

```
// connect messages state-change signal to our slot, to get notification
connect(&m_MessageServiceAction, SIGNAL(stateChanged(QMessageServiceAction::State)),
this, SLOT(messageStateChanged(QMessageServiceAction::State)));

void send_sms::QtSendSMSMessage(QString address, QString body) {
    // Prepare QMessage with address & body
    QMessage message;
    message.setType(QMessage::Sms);
    message.setTo(QMessageAddress(address, QMessageAddress::Phone));
    message.setBody(body);

    if (iMessageServiceAction.send(message)) {
```


(C) Copyright Nokia 2014. All rights reserved.

```
        sendId = message.id();
        }
    else {
        // message sending failed
        }
    }

void send_sms::messageStateChanged(QMessageServiceAction::State s) {
    if (s == QMessageServiceAction::Successful) {
        // message send successful
        }
    else {
        // message sending failed
        }
    }
```

Classes

- QMessageServiceAction
- QMessageId

Reference links

- Qt - cross-platform application and UI framework
- Qt Mobility API
- New Qt APIs Beta - Mobility Project
- SDK help

(C) Copyright Nokia 2014. All rights reserved.