Add contact to contact database using Qt Mobility

Overview

This code snippets shows how to add contact to selected(symbian) contact manager. This code example can easily extended to add contact to any other contact manager by just iterating through available contact maangers and selecting appropriate one.

This snippet requires ReadDeviceData and WriteDeviceData capabilities. Self-signing is not possible and atleast Developer certificate is needed.

Prerequisite

- Install QtMobility APIs:Setting up environment for Qt Mobility API

Headers required

```
#include <qtcontacts.h>
```

.pro file

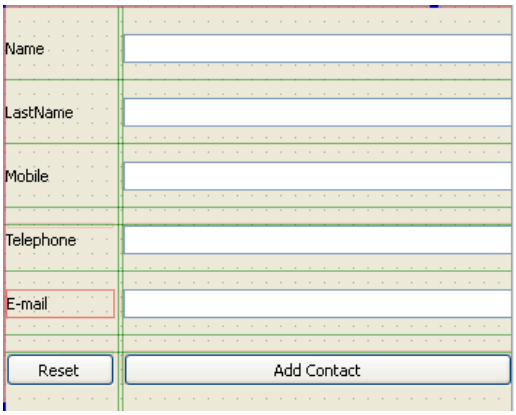
Modify your .pro file as follows.

```
INCLUDEPATH += D:/QtMobility/src/contacts \
INCLUDEPATH += D:/QtMobility/src/contacts/filters \
INCLUDEPATH += D:/QtMobility/src/contacts/requests \
INCLUDEPATH += D:/QtMobility/src/contacts/details
INCLUDEPATH += D:/QtMobility/src/global

CONFIG += mobility
MOBILITY = contacts

symbian::TARGET.CAPABILITY = ReadUserData \
    WriteUserData \
    ReadDeviceData \
    WriteDeviceData
```

UI designer(.ui) file



Header file

```
/* AddContact .h */
#ifndef ADDCONTACT_H
#define ADDCONTACT_H

#include <QtGui/QMainWindow>
#include "ui_AddContact.h"
#include "qtcontacts.h"

QTM_USE_NAMESPACE

class AddContact : public QMainWindow
{

    Q_OBJECT

public:
 AddContact(QWidget *parent = 0);
    ~AddContact();
```


(C) Copyright Nokia 2013. All rights reserved.

```
public slots:
 void resetContactDetail();
 void addNewContact();

private:
    Ui::AddContactClass ui;
    QContactManager *contactManager;
};

#endif // ADDCONTACT_H
```

Source file

```
/* AddContact.cpp */
#include "AddContact.h"
#include <QMessageBox>

AddContact::AddContact(QWidget *parent)
    : QMainWindow(parent), contactManager(0)
{
 ui.setupUi(this);
 setWindowTitle("Add contacts");
 connect(ui.pushButtonAddContact, SIGNAL(clicked()), this, SLOT(addNewContact()));
 connect(ui.pushButtonReset, SIGNAL(clicked()), this, SLOT(resetContactDetail()));
}

void AddContact::addNewContact()
{
 if( (ui.inputName->text()).isEmpty() && (ui.inputLastName->text()).isEmpty() &&
  (ui.inputMobile->text()).isEmpty() && (ui.inputTelephone->text()).isEmpty() &&
  (ui.inputEmail->text()).isEmpty())
 {
  QMessageBox::information(this, "Contacts!", "Enter contact information!");
  return;
 }

 if(!contactManager)
 {
  /* Adding contact to "symbian" contact manager */
  contactManager = new QContactManager("symbian");

  /* You can also add contact to selected contact managet by selecting
   * appropriate contact manager from available contact managers.
   */
  /*
  QStringList availableManagers = QContactManager::availableManagers();
  contactManager = new QContactManager(availableManagers.at(2));
  */
 }

 /* initialise contact details from user */
 QContact curr;
 QContactName name = curr.detail(QContactName::DefinitionName);
 QContactPhoneNumber phone = curr.detail(QContactPhoneNumber::DefinitionName);
 QContactPhoneNumber telephone = curr.detail(QContactPhoneNumber::DefinitionName);
 QContactEmailAddress email = curr.detail(QContactEmailAddress::DefinitionName);

 name.setFirst(ui.inputName->text());
 name.setLast(ui.inputLastName->text());

 phone.setContexts(QContactDetail::ContextHome);
 phone.setSubTypes(QContactPhoneNumber::SubTypeMobile);
 phone.setNumber(ui.inputMobile->text());

 phone.setContexts(QContactDetail::ContextWork);
 telephone.setSubTypes(QContactPhoneNumber::SubTypeLandline);
 telephone.setNumber(ui.inputTelephone->text());

 email.setEmailAddress(ui.inputEmail->text());

 /* save details to contact object */
 curr.saveDetail(&name);
 curr.saveDetail(&phone);
 curr.saveDetail(&telephone);
 curr.saveDetail(&email);

 /* save contact detail to contact manager */
 bool success = contactManager->saveContact(&curr);
 if (success)
 {
  QMessageBox::information(this, "Success!", "Contact " + ui.inputName->text() + " saved successfully!");
  resetContactDetail();
 }
 else
 {
  QMessageBox::information(this, "Failed!", "Error in saving contact!");
 }
}

void AddContact::resetContactDetail()
{
 ui.inputName->clear();
 ui.inputLastName->clear();
 ui.inputMobile->clear();
 ui.inputTelephone->clear();
 ui.inputEmail->clear();
}

AddContact::~AddContact()
{
 delete contactManager;
}
```

Post-conditions

The code snippet is expected to add contact to symbian contact manager.

(C) Copyright Nokia 2013. All rights reserved.

Download Code Example

- Working Code Example is available to download from here.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.