How to use QCompass sensor

This code snippet shows how to use Qt's QCompass sensor and display sensor data on the screen.

Project File (.pro file)

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

```
CONFIG += mobility
MOBILITY += sensors
```

- here is the full project(.pro) file.

```
QT       += core gui

TARGET = QtCompasSensor
TEMPLATE = app


SOURCES += main.cpp\
        qtcompassensor.cpp

HEADERS  += qtcompassensor.h

CONFIG += mobility
CONFIG += debug
MOBILITY = sensors

symbian {
    TARGET.UID3 = 0xe00fb3c0
    # TARGET.CAPABILITY +=
    TARGET.EPOCSTACKSIZE = 0x14000
    TARGET.EPOCHEAPSIZE = 0x020000 0x800000
}
```

Header

- Add the Qt Mobility namespace to use Qt Mobiltiy in header file

```
QTM_USE_NAMESPACE
```

Here is the qtcompassensor.h header file.

```
#include <QtGui/QMainWindow>
#include <QGraphicsView>
#include <QBasicTimer>
#include <QTimer>
#include <QCompass> // Add Sensor Class

// add Qt Mobility Project Namespace
QTM_USE_NAMESPACE
```


(C) Copyright Nokia 2014. All rights reserved.

```
class QGraphicsRectItem;
class QGraphicsSimpleTextItem;

class QtCompasSensor : public QGraphicsView
{
    Q_OBJECT

public:
    QtCompasSensor(QWidget *parent = 0);

protected:
    bool event(QEvent *event);

private:
    void updateCompasSensor();

private:
    QGraphicsRectItem *Rect;
    QGraphicsSimpleTextItem *text;
    QCompass *CompasSensor;
    QBasicTimer Timer;
    QTimer inactiveTimer;
};
```

Source

Here is the qtcompassensor.cpp source file.

```
#include "qtcompassensor.h"
#include <QGraphicsRectItem>
#include <QGraphicsSimpleTextItem>
#include <QResizeEvent>

// include timer event
#include <QTimerEvent>
#include <QDebug>

QtCompasSensor::QtCompasSensor(QWidget *parent)
    : QGraphicsView(parent), Rect(0), text(0), CompasSensor(0)
{
    QGraphicsScene *scene = new QGraphicsScene(this);
    setScene(scene);

    Rect = new QGraphicsRectItem();

    text = new QGraphicsSimpleTextItem(Rect);
    text->setBrush(Qt::gray);
    scene->addItem(Rect);

    CompasSensor = new QCompass(this);
    inactiveTimer.setSingleShot(true);

    if (!Timer.isActive())
```

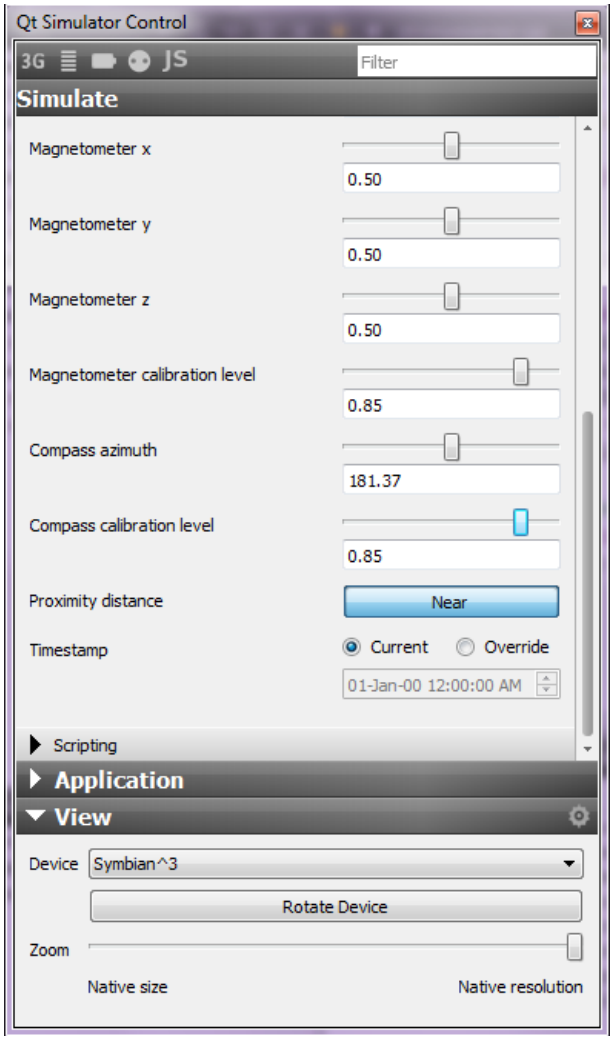

(C) Copyright Nokia 2014. All rights reserved.

```
        Timer.start(20, this);

    // start the sensor
    if (!CompasSensor->isActive())
        CompasSensor->start();

    if (!CompasSensor->isActive())
    {
        qDebug() << "Compas Sensor didn't start!" << endl;
    }

    setHorizontalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
    setVerticalScrollBarPolicy(Qt::ScrollBarAlwaysOff);
}

bool QtCompasSensor::event(QEvent *event)
{
    switch (event->type()) {
    case QEvent::Timer:
        if (static_cast<QTimerEvent*>(event)->timerId() == Timer.timerId())
            updateCompasSensor(); // update the sensor data
        break;
    default:
        break;
    }
    return QGraphicsView::event(event);
}

void QtCompasSensor::updateCompasSensor()
{
    QCompassReading *reading = CompasSensor->reading();
    qreal azi = 0.0f;
    qreal calibrationLevel = 0.0f;
    if (reading) {
        azi = reading->azimuth();
        calibrationLevel = reading->calibrationLevel();

        // set the text and display it on the screen
        QLocale c(QLocale::C);
        text->setText("azimuth: " + c.toString(azi) + "\n\nCompass Sensor Calibration
Level: " + c.toString(calibrationLevel));

     }
}
```

Postconditions

Here is the QtCompassSensor Application's Screenshot. You can change the Compass Sensor data like azimuth and Calibration Level from Qt Simulator.

(C) Copyright Nokia 2014. All rights reserved.



Example Application

Example application can be found on the following link File:QtCompassSensorExample.zip.

Related Articles

- Turning phone into magnetic compass using Qt mobility project
- Accelerometer Sensor Example Using QtMobility Project
- QtMagnetometer Sensor Example Using QtMobility Project
- Qt Mobility 1.2

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.