Como executar o navegador Web, RealPlayer, e outras aplicações, em Qt

Introdução

Frequentemente nós executamos outras aplicações através da nossa aplicação. Por exemplo, precisamos executar um navegador para abrir uma página da web ou executar um player pra abrir um link de vídeo. Do mesmo modo, algumas vezes necessitamos executar outras aplicações através da nossa aplicação. Em Qt, a classe QProcess é usada para executar programas externos e se comunicar com eles.

A classe QDesktopServices provê métodos para acessar serviços comuns do desktop, como navegadores, players, cliente de e-mail, etc.

QDesktopServices possui o método estático *openUrl()* que abre a URL dada no próprio navegador para o ambiente desktop dos usuários e retorna *true* se tudo ocorreu com sucesso ou, caso contrário, *false*. Se a URL é uma referência para um *arquivo local*,(isto é, o esquema de URl é o *arquivo*), então ele será aberto com uma aplicação adequada do navegador Web. Por exemplo, você pode usar a URL "file:///c:/data/abc.png" para abrir abc.png na aplicação padrão. Similarmente você pode reproduzir áudio/vídeo.

O trecho de código pode ser auto-assinado. Uma vez que ele não usa nenhuma API que requer assinatura de desenvolvedor/certificado.

Pré-requisitos

- Baixe e instale a versão atual do Qt para Symbian segundo as instruções deste artigo.

Lançando o navegador Web

Cabeçalhos requeridos

```
#include <QDesktopServices>
#include <QUrl>
```

Código fonte

```
//openUrl é um método estático da classe QDesktopServices. Assim, não é necessário criar uma instância desta classe
//Abre a URL através do browser
QDesktopServices::openUrl(QUrl(tr("http://www.developer.nokia.com/Community/Discussion/forumdisplay.php?40-General-Symbian-C")));
```

Lançando o RealPlayer

Cabeçalhos requeridos

```
#include <QDesktopServices>
#include <QUrl>
```

Código fonte

```
//openUrl é um método estático da classe QDesktopServices. Assim, não é necessário criar uma instância desta classe
//para abrir o arquivo de video no realplayer, adicione o endereço completo abaixo.
QDesktopServices::openUrl(QUrl(tr("file:///e:/Videos/Nokia 5800 XpressMusic.mp4")));
```

Lançando um cliente de e-mail

Cabeçalhos requeridos

```
#include <QDesktopServices>
#include <QUrl>
```

Código fonte

```
//openUrl é um método estático da classe QDesktopServices. Assim, não é necessário criar uma instância desta classe
//Abre um cliente de email
QDesktopServices::openUrl(QUrl(tr("mailto:ratang.gir@gmail.com?subject=Testing feedback&body=HelloFriend")));
```


(5) Copyright Nokia 2013. All rights reserved.

Lançando outras aplicações

Cabeçalhos requeridos

```
#include <QProcess>
```

O sinal *error()* de QProcess é emitido quando um erro ocorre com o processo. O erro especificado descreve o erro que ocorreu.

Cabeçalho de arquivo

```
void processError(QProcess::ProcessError err);

QProcess process;
```

Código fonte

```
QtAppLauncher::QtAppLauncher(QWidget *parent)
    : QMainWindow(parent)
{
//connect error() to slot processError() to get error, if occurs.
QObject::connect( &process, SIGNAL(error(QProcess::ProcessError)),
                        this, SLOT(processError(QProcess::ProcessError)));

//passa o nome do executável que você deseja iniciar a execução. Instalei um ''helloworld'', assim, passe helloworld.
process.start("helloworld");
}

//Captura erros, se existir
void QtAppLauncher::processError(QProcess::ProcessError err)
{
 switch(err)
 {
 case QProcess::FailedToStart:
  QMessageBox::information(0,"FailedToStart","FailedToStart");
  break;
 case QProcess::Crashed:
  QMessageBox::information(0,"Crashed","Crashed");
  break;
 case QProcess::Timedout:
  QMessageBox::information(0,"FailedToStart","FailedToStart");
  break;
 case QProcess::WriteError:
  QMessageBox::information(0,"Timedout","Timedout");
  break;
 case QProcess::ReadError:
  QMessageBox::information(0,"ReadError","ReadError");
  break;
 case QProcess::UnknownError:
  QMessageBox::information(0,"UnknownError","UnknownError");
  break;
 default:
  QMessageBox::information(0,"default","default");
  break;
 }
}
```

Conclusões

É esperado que o trecho de código lance um navegador, o RealPlayer e uma aplicação HelloWord.

Código exemplo

- Faça o download do código exemplo.

(5) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.