Implementing a custom CAknQueryDialog dialog

Overview

This snippet shows how to implement a custom dialog. In this case the dialog acts like an application splash screen. The dialog closes itself after two seconds and the application continues to run.

This snippet can be self-signed.

MMP file

The following capabilities and libraries are required:

```
CAPABILITY None

LIBRARY avkon.lib
LIBRARY eikcoctl.lib
LIBRARY eikdlg.lib
LIBRARY euser.lib
```

Header file

```
#include <aknquerydialog.h>
#include <e32std.h>
#include <e32base.h>
#include <eikbtgpc.h>

class CSplashScreenDialog : public CAknQueryDialog
    {
    public: // Construction
        CSplashScreenDialog();
        virtual ~CSplashScreenDialog();
        void PreLayoutDynInitL();
        void PostLayoutDynInitL();
        TBool OkToExitL(TInt aButtonId);

    private: // From base classes
        void SizeChanged();
        TInt CountComponentControls() const;
        CCoeControl* ComponentControl(TInt aIndex) const;
        void Draw( const TRect& /*aRect*/ ) const;
        void SetSizeAndPosition(const TSize &aSize);

    private: // Internal methods
        void CalculatePositionAndSize();
        static TInt HideDialog(TAny* aAny);

    private: // Data
        CPeriodic*          iPeriodic;
    };
```


(C) Copyright Nokia 2014. All rights reserved.

Source file

```
#include "csplashscreendialog.h"

const TInt KShowTime(2000000);

CSplashScreenDialog::CSplashScreenDialog()
    {
    // Closes this dialog after KShowTime 2s
    TRAPD(err,
            iPeriodic = CPeriodic::NewL(CActive::EPriorityIdle);
    );
    if (!err)
        {
        iPeriodic->Start(KShowTime, KShowTime, TCallBack(HideDialog, this));
        }
    }

CSplashScreenDialog::~CSplashScreenDialog()
    {
    if (iPeriodic)
        {
        iPeriodic->Cancel();
        }
    delete iPeriodic;
    }

void CSplashScreenDialog::PreLayoutDynInitL()
    {
    CAknQueryDialog::PreLayoutDynInitL();
    SetEditableL(EFalse);
    }

void CSplashScreenDialog::PostLayoutDynInitL()
    {
    // No CBA buttons
    CEikButtonGroupContainer* cba = CEikButtonGroupContainer::Current();
    cba->MakeVisible(EFalse);

    // Calculate position of the dialog
    CalculatePositionAndSize();
    }

TBool CSplashScreenDialog::OkToExitL(TInt /*aButtonId*/)
    {
    return ETrue;
    }

TInt CSplashScreenDialog::CountComponentControls() const
    {
    // No child components in this example
    return(0);
    }

CCoeControl* CSplashScreenDialog::ComponentControl(TInt /*aIndex*/) const
    {
```


(C) Copyright Nokia 2014. All rights reserved.

```
    // No child components in this example
    return NULL;
    }

void CSplashScreenDialog::CalculatePositionAndSize()
    {
    // Set full screen
    SetExtentToWholeScreen();
    }

void CSplashScreenDialog::SetSizeAndPosition(const TSize &aSize)
    {
    CAknQueryDialog::SetSizeAndPosition( aSize );
    CalculatePositionAndSize();
    }

void CSplashScreenDialog::SizeChanged()
    {
    // Update screen
    DrawNow();
    }

void CSplashScreenDialog::Draw( const TRect& /*aRect*/ ) const
    {
    // Get the standard graphics context
    CWindowGc& gc = SystemGc();
    gc.SetPenStyle( CGraphicsContext::ENullPen );
    gc.SetBrushColor( KRgbBlack);
    gc.SetBrushStyle( CGraphicsContext::ESolidBrush );

    // Clear screen
    gc.Clear(Rect());
    gc.SetBrushStyle( CGraphicsContext::ENullBrush );

    // Draw round rect
    gc.SetPenStyle( CGraphicsContext::ESolidPen );
    gc.SetPenColor(KRgbWhite);
    gc.DrawRect(Rect());

    // TODO: Implement what you need
    }

TInt CSplashScreenDialog::HideDialog(TAny* aAny)
    {
    CSplashScreenDialog* self = static_cast<CSplashScreenDialog*>( aAny );

    // Cancel CPeriodic timer
    self->iPeriodic->Cancel();

    // Closes itself
    TRAP_IGNORE(
            self->TryExitL(1);
            );
    return KErrNone;
    }
```


(C) Copyright Nokia 2014. All rights reserved.

Custom dialog resurce

```
RESOURCE DIALOG r_custom_dialog
    {
    flags = EGeneralQueryFlags;
    buttons = R_AVKON_SOFTKEYS_EMPTY;
    }
```

Running the custom dialog

```
iSplashScreenDialog = new (ELeave) CSplashScreenDialog();
iSplashScreenDialog->ExecuteLD(R_CUSTOM_DIALOG);
// Execution continues here after dialog has hided itself
iSplashScreenDialog = NULL;
```

Postconditions

The custom dialog is shown for two seconds.

(C) Copyright Nokia 2014. All rights reserved.