Como usar Cimbalino Windows Phone Toolkit Camera - CameraCaptureService

Este artigo tem como objetivo mostrar como usar Cimbalino Windows Phone Toolkit Camera – CameraCaptureService.

Introdução

Cimbalino Windows Phone Toolkit é um conjunto de itens uteis e poderosos para ajudar na implementação de aplicações Windows Phone. Projeto base do toolkit contendo serviços para implementação do padrão de MVVM, conversores, classes auxiliares, métodos de extensões;

Cimbalino.Phone.Toolkit.Camera - Projeto do toolkit contendo serviços para implementação do padrão de MVVM que faz uso da câmera.

- ICameraCaptureService - Representa uma interface para um serviço capaz de lançar a aplicação Câmara. A implementaçãp é CameraCaptureService

Note: é preciso definir a capacidade ID_CAP_ISV_CAMERA no ficheiro manifest.

Contruindo o exemplo

O código fonte pode ser obtido em Exemplo CameraCaptureService (github).

Os pacotes estão disponíveis em Nuget Package Manager (para ambos os "targets") e podem ser instalados, para mais detalhes consultar: Como instalar os pacotes Cimbalino Windows Phone Toolkit. O exemplo deste artigo usa MVVM Light para ajudar na implementação do padrão MVVM, para mais detalhes consultar:

- Como instalar os pacotes de MVVM Light Toolkit para Windows Phone
- Como usar MVVM Light Toolkit para Windows Phone

Registando os serviços

Devemos começar por registar cada serviço no ViewModelLocator, como podemos ver de seguida:

```
/// This class contains static references to all the view models in the
/// application and provides an entry point for the bindings.
public class ViewModelLocator
{
    /// Initializes a new instance of the ViewModelLocator class.
    public ViewModelLocator()
    {
        ServiceLocator.SetLocatorProvider(() => SimpleIoc.Default);
```

 (C) Copyright Nokia 2014. All rights reserved.

```
            if (!SimpleIoc.Default.IsRegistered<ICameraCaptureService>())
            {
                SimpleIoc.Default.Register<ICameraCaptureService, CameraCaptureService>();
            }

            SimpleIoc.Default.Register<MainViewModel>();
        }

        public MainViewModel MainViewModel
        {
            get
            {
                return ServiceLocator.Current.GetInstance<MainViewModel>();
            }
        }

        public static void Cleanup()
```

http://developer.nokia.com/community/wiki/Como_usar_Cimbalino_Windows_Phone_Toolkit_Camera_-_CameraCaptureService (C) Copyright Nokia 2014. All rights reserved.

```
        {
            // TODO Clear the ViewModels
        }
    }
```

Implementando a ViewModel

Em seguinda devemos implementar o MainViewModel, como podemos ver de seguida:

```
    /// This class contains properties that the main View can data bind to.
    public class MainViewModel : ViewModelBase
    {
        /// The camera capture service
        private readonly ICameraCaptureService _cameraCaptureService;

        /// The image
        private ImageSource _image;

        /// Initializes a new instance of the MainViewModel class.
        public MainViewModel(ICameraCaptureService cameraCaptureService)
        {
```

 (C) Copyright Nokia 2014. All rights reserved.

```
            _cameraCaptureService = cameraCaptureService;

            CameraCaptureCommand = new RelayCommand(CameraCapture);

        }

        /// Gets the camera capture command.

        public ICommand CameraCaptureCommand { get; private set; }

        /// Gets or sets the image.

        public ImageSource Image

        {

            get

            {

                return _image;

            }

            set

            {

                Set("Image", ref _image, value);

            }
```

http://developer.nokia.com/community/wiki/Como_usar_Cimbalino_Windows_Phone_Toolkit_Camera_-_CameraCaptureService (C) Copyright Nokia 2014. All rights reserved.

```
        }

        /// Cameras the capture.
        private void CameraCapture()
        {
            Image = null;
            _cameraCaptureService.Show(CameraCapetureResult);
        }

        /// Cameras the capture result.
        private async void CameraCapetureResult(Microsoft.Phone.Tasks.PhotoResult photoResult)
        {
            if(photoResult.ChosenPhoto!=null)
            {
                var bitmapImage = new BitmapImage();
                bitmapImage.SetSource(photoResult.ChosenPhoto);
                Image = bitmapImage;
            }
```

http://developer.nokia.com/community/wiki/Como_usar_Cimbalino_Windows_Phone_Toolkit_Camera_-_CameraCaptureService (C) Copyright Nokia 2014. All rights reserved.

```
        }
```

```
    }
```

Implementando a View

A **MainPage.xaml** será algo do género:

```
<phone:PhoneApplicationPage
    x:Class="CimbalinoSample.MainPage"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:phone="clr-namespace:Microsoft.Phone.Controls;assembly=Microsoft.Phone"
    xmlns:shell="clr-namespace:Microsoft.Phone.Shell;assembly=Microsoft.Phone"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d"
    FontFamily="{StaticResource PhoneFontFamilyNormal}"
    FontSize="{StaticResource PhoneFontSizeNormal}"
    Foreground="{StaticResource PhoneForegroundBrush}"
    DataContext="{Binding MainViewModel, Source={StaticResource Locator}}"
    SupportedOrientations="Portrait" Orientation="Portrait"
```

 (C) Copyright Nokia 2014. All rights reserved.

```
    shell:SystemTray.IsVisible="True">

    <!--LayoutRoot is the root grid where all page content is placed-->
    <Grid x:Name="LayoutRoot" Background="Transparent">
        <Grid.RowDefinitions>
            <RowDefinition Height="Auto"/>
            <RowDefinition Height="*"/>
        </Grid.RowDefinitions>

        <!--TitlePanel contains the name of the application and page title-->
        <StackPanel x:Name="TitlePanel" Grid.Row="0" Margin="12,17,0,28">
            <TextBlock Text="Cimbalino Sample" Style="{StaticResource PhoneTextTitle2Style}" Margin="12,0"/>
            <TextBlock Text="CameraCaptureService and PhotoCameraService"
                       TextWrapping="Wrap"
                       Margin="9,-7,0,0" Style="{StaticResource PhoneTextTitle2Style}"/>
        </StackPanel>

        <!--ContentPanel - place additional content here-->
```

 (C) Copyright Nokia 2014. All rights reserved.

```
        <Grid x:Name="ContentPanel" Grid.Row="1" Margin="12,0,12,0">
```

```
            <Image Source="{Binding Image}" Stretch="Uniform" Margin="0,242,0,0"
/>
```

```
            <Button Content="Camera Capture" Command="{Binding
CameraCaptureCommand}" Height="237" VerticalAlignment="Top"/>
```

```

```

```
        </Grid>
```

```
    </Grid>
```

```
</phone:PhoneApplicationPage>
```

http://developer.nokia.com/community/wiki/Como_usar_Cimbalino_Windows_Phone_Toolkit_Camera_-_CameraCaptureService (C) Copyright Nokia 2014. All rights reserved.