

Advanced MessageBox for Windows Phone

This article shows how to create a customised messagebox: asynchronous, custom button, no beep, etc.

Introduction



The standard phone message boxes is very easy to use, and is created with just a single line:

```
Message.Show("Hello!");
```

But these message boxes aren't easy to customize.

We will learn in this article how to :

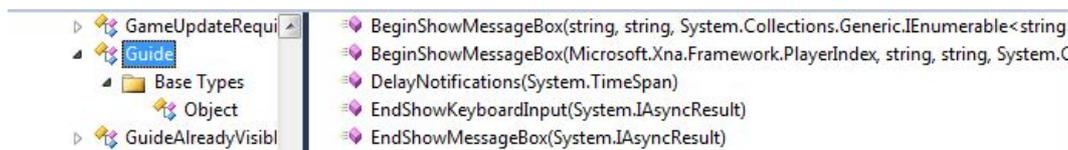
- create an asynchronous native message box
- customize buttons
- remove beep

Accessing advanced features through XNA

There is a way to access advanced features of the native messagebox through one back door using XNA.

 Tip: You can access XNA framework in a Windows Phone 8 project, but only for some namespaces (you can't make full-XNA games for Windows Phone 8). The list of allowed namespaces is in [XNA Framework and Windows Phone 8 development](#).

If you look at the `Microsoft.Xna.Framework.GameServices` assembly, you will notice that the `Guide` object has the following methods :



`BeginShowMessageBox()` method displays a native and asynchronously messagebox with parameters:

- `title`: Title of the message box
- `text`: text to be displayed in the message box
- `buttons`: Legends associated with the buttons on the message box. The maximum number of buttons is two.
- `focusButton`: 0-based index that defines the button highlighted.
- `icon`: Type icon in the message box.
- `callback`: method to call when the asynchronous operation is complete.
- `STATE`: The unique user-created that identifies this request.

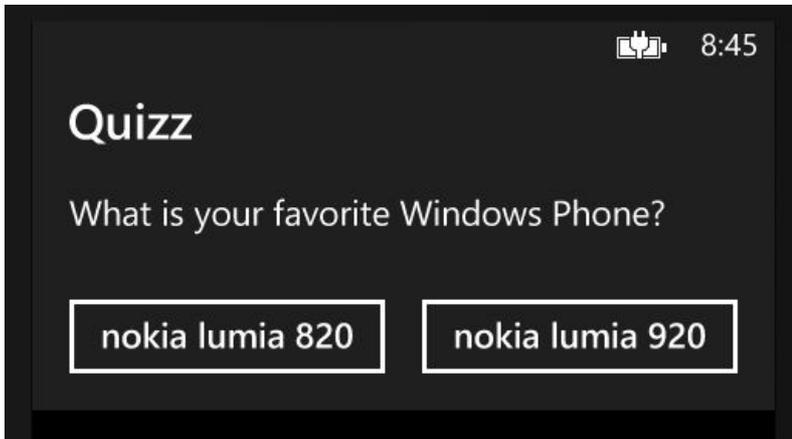
Title and text match mutually the Silverlight parameters `caption` and `messageboxtext` of Silverlight.

One difference here: the text can not exceed 256 characters, otherwise an exception is thrown.

Customize buttons

Buttons is here much more advanced than our Silverlight MessageBox, we can specify the button text, for example, we can modify the "ok" and "cancel" text like that :

```
Microsoft.Xna.Framework.GamerServices.Guide.BeginShowMessageBox(
    "Quizz",
    "What is your favorite Windows Phone?",
    new string[] { "Nokia Lumia 820", "Nokia Lumia 920" },
    0,
    Microsoft.Xna.Framework.GamerServices.MessageBoxIcon.Alert,
    null,
    null);
```



This offers us a lot of new possibilities, but take care to localize your buttons.

How to make synchronous messagebox?

Be able to launch a asynchronous dialog is pretty cool, but in most cases, we expect it to be synchronous. In order to do this, simply retrieve the result of the asynchronous call and wait for its execution.

```

IAsyncResult result = Microsoft.Xna.Framework.GamerServices.Guide.BeginShowMessageBox(
    "Quizz",
    "What is your favorite Windows Phone?",
    new string[] { "Nokia Lumia 820", "Nokia Lumia 920" },
    0,
    Microsoft.Xna.Framework.GamerServices.MessageBoxIcon.None,
    null,
    null);

result.AsyncWaitHandle.WaitOne();

```

We have now a native, synchronous and customizable message box !

Test result of the dialog

If you want to test the result of your dialog box to know which button has been clicked, simply call the method `EndShowMessageBox`:

```

int? choice = Microsoft.Xna.Framework.GamerServices.Guide.EndShowMessageBox(result);
if(choice.HasValue)
{
    if(choice.Value==0)
    {
        //user clicks the first button
    }
}

```

How to remove the messagebox sound?

To remove the beep or the vibration, just change the icon... It's not logical, but that's how it works.

XNA is a common platform between Windows, Xbox and Windows Phone. On windows, we are used to display icons like 'warning', 'alert', etc. on the left of the dialog box but this isn't ideal for mobile screen. The distinction between a normal dialog and a warning/alert dialog box is therefore the sound / vibration. To remove the sound, we just have to set the `type` to `None`.

```

Microsoft.Xna.Framework.GamerServices.Guide.BeginShowMessageBox(

```

```
"Quizz",  
"What is your favorite Windows Phone?",  
new string[] { "Nokia Lumia 820", "Nokia Lumia 920" },  
0,  
Microsoft.Xna.Framework.GamerServices.MessageBoxIcon.None,  
null,  
null);
```

No sound will be launch.

Conclusion

- Add reference `Microsoft.Xna.Framework.GameServices` (not necessary with Windows Phone 8 project)
- Write the following code:

```
IAsyncResult result =  
Microsoft.Xna.Framework.GamerServices.Guide.BeginShowMessageBox(  
    "Quizz",  
    "What is your favorite Windows Phone?",  
    new string[] { "Nokia Lumia 820", "Nokia Lumia 920" },  
    0,  
    Microsoft.Xna.Framework.GamerServices.MessageBoxIcon.None,  
    null,  
    null);  
  
result.AsynchWaitHandle.WaitOne();  
  
int? choice = Microsoft.Xna.Framework.GamerServices.Guide.EndShowMessageBox(result);  
if(choice.HasValue)  
{  
    if(choice.Value==0)  
    {  
        //User clicks on the first button  
    }  
}
```