

4. HOW TO ADD APPLICATION BAR IN WINDOWS PHONE 7

In every phone application bar play very important role to navigate application very smooth. Some time application requirement is to use common functionality in entire application. In this case we can use the application bar. In windows phone 7 have a facility to add application bar using xaml code. In this article I am going to explain how to add application bar icons, its event and its navigation for the better user experience.

Where is application bar in windows phone applications?

Please see the below attached code with pictorial representation in pdf document.



Let us start to work with application bar in Windows Phone.

Step-1: Create a Windows Phone 7 project.

Step-2: Now add an icon directory in the project; after added icon directory in the project we are putting required icons in this directory and select all icons and press F4 button then opened properties window then go to Build Action and choose Content.

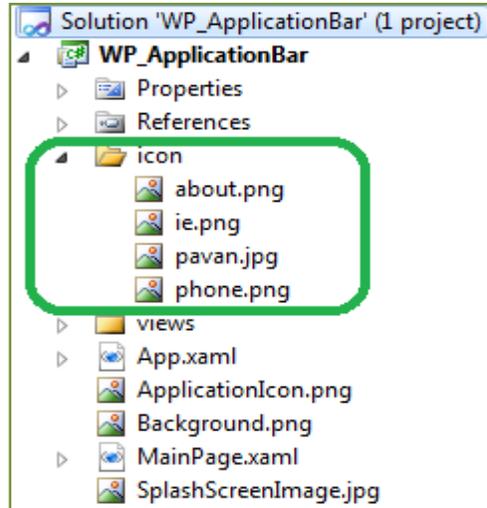
Step-3: Now open "App.xaml" file in design mode. In the xaml file there is an application resource tag, in this section we need to add shell:Application tag and set IsVisible property to True. See below xaml code snippet.

[code]

```
<Application.Resources>
  <shell:ApplicationBar x:Key="MainAppBar" IsVisible="True">
    </shell:ApplicationBar>
</Application.Resources>
```

[/code]

Step-4: Now in the shell:Application tag we are adding required application bar icons as per application requirement, here I have added three icons like web, phone and about me Icons and its event which will be automatically added in App.xaml.cs file. See below code snippet for add the buttons in application bar.



[code]

```
<Application.Resources>
  <shell:ApplicationBar x:Key="MainAppBar" IsVisible="True">
    <shell:ApplicationBar.Buttons>
      <shell:ApplicationBarIconButton Text="Web" IconUri="icon\ie.png"
Click="ApplicationBarIconWebButton_Click"/>
      <shell:ApplicationBarIconButton Text="Phone" IconUri="icon\phone.png"
Click="ApplicationBarIconPhoneButton_Click" />
      <shell:ApplicationBarIconButton Text="About Me" IconUri="icon\about.png"
Click="ApplicationBarIconAboutButton_Click"/>
    </shell:ApplicationBar.Buttons>
  </shell:ApplicationBar>
</Application.Resources>
```

[/code]

In the code behind of App.xaml file event is created automatically when we added events in xaml code. See below code snippet for the event creation.

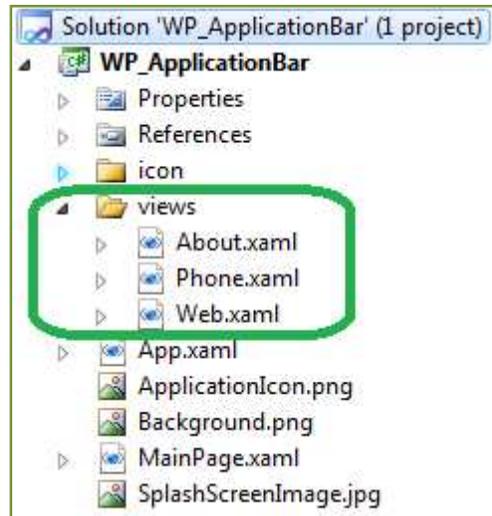
[code]

```
private void ApplicationBarIconWebButton_Click(object sender, EventArgs e)
{
}

private void ApplicationBarIconPhoneButton_Click(object sender, EventArgs e)
{
}
```

```
private void ApplicationBarIconAboutButton_Click(object sender, EventArgs e)
{
}
[/code]
```

Step-5 Now we are going to navigate the application using application bar button's event handler; for this navigation. I have added a views directory in the project and in views directory I have added three xaml pages like About.xaml, Phone.xaml and Web.xaml.



About.xaml:- This page is navigate throw the application bar button when the user click on the “About” icon button. This page contents the author information.

Phone.xaml:- This page is designed for making call programmatically. For the time being I have not added code for the same, in this series will discuss it in launcher part.

Web.xaml:- This page contents the web browser control. This page itself explains how to use web browser control in windows phone 7.

Once we designed these pages, then we go ahead and added the navigation code in App.xaml.cs for navigation these pages throw the application bar icon buttons.

Step-6: Now add below code snippet for the page navigation to appropriate event handler like web icon's event has web page navigation code. See below code snippet.

[code]

```
private void ApplicationBarIconWebButton_Click(object sender, EventArgs e)
{
    PhoneApplicationFrame root = Application.Current.RootVisual as PhoneApplicationFrame;
    root.Navigate(new Uri("/views/Web.xaml", UriKind.Relative));
}
```

```

private void ApplicationBarIconPhoneButton_Click(object sender, EventArgs e)
{
    PhoneApplicationFrame root = Application.Current.RootVisual as PhoneApplicationFrame;
    root.Navigate(new Uri("/views/Phone.xaml", UriKind.Relative));
}

private void ApplicationBarIconAboutButton_Click(object sender, EventArgs e)
{
    PhoneApplicationFrame root = Application.Current.RootVisual as PhoneApplicationFrame;
    root.Navigate(new Uri("/views/About.xaml", UriKind.Relative));
}

[/code]

```

Code explanation:

The PhoneApplicationFrame control allows developer to navigate one screen to other screens. The PhoneApplicationFrame is a top-level container control of windows phone application.

Here I have created root instance of PhoneApplicationFrame and initialized with application current visual root. This root instance has a Navigate method that allows to windows phone page navigate using relative URI of the page.

Step-7: Now compile and run the application. Test the navigation by clicking on application bar buttons. To see the screenshots, refer the attached article with pictorial representation.



Now click on the about button on application bar then about me page is navigated see blow screenshot.



Thanks you☺!