Week3 – Navigation page demo

There is a variety of ways to navigate through Xamarin.Forms, on which more can be found in your eBook – **chapter 24: Page navigation**. You can find various demo's in this repository to demonstrate different page types and navigation.

In this document, we will explain how to use a **Navigation Page**. This page type allows you to navigate from one page to another, and back.

Page type(s)

A navigation page is a common **ContentPage** that will be wrapped as a Navigation Page.

Let's say your contentpage is called MainPage, than you can turn it into a navigation page like this:

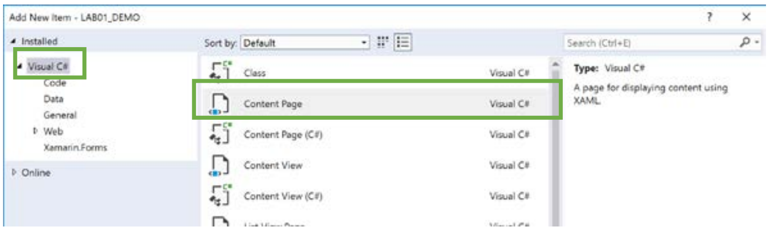
```
        MainPage = new NavigationPage( new MainPage() );
```

How to create a navigation page

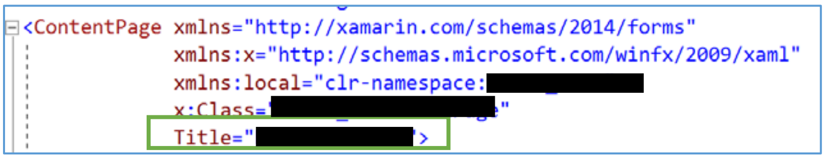
1. **Create a content page** where you want to navigate from (and another where you want to navigate to).
2. **Turn the content page into a navigation page** by simply changing the way it's created (for your main page, this will be in app.xml.cs).

1. Create a content page

- Create a folder '**View**' in the PCL project
- Inside this folder, add a new item and name it:



- Add your content inside the <ContentPage.Content> tags.
- Make sure you set the **Title** property of the ContentPage:



```
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             xmlns:local="clr-namespace:[redacted]
             x:Class=[redacted]"
             Title="[redacted]">
```

2. Turn a ContentPage into a NavigationPage

- Go to App.xaml.cs
- In the constructor, you will find:

```
        MainPage = new LAB01_DEMO.MainPage();
```

- Turn the page into a navigation page by changing this line of code to:

```
        MainPage = new NavigationPage( new LAB01_DEMO.MainPage() );
```

Navigating using a NavigationPage

The Push(Modal)Async function uses the power of asynchronous programming. We will see the concept of this further in detail later in the Device Programming II course.

For the differences between modal and modeless pages, we refer to the theory lesson.

a. Navigation without data exchange

- Go to the code behind of the page you wish to navigate **from** (eg.: MainPage.xaml.cs)
- You can now navigate like this:

```
        Navigation.PushAsync(new PageName()); //PageName = name of your page
        Navigation.PushModalAsync(new PageName()); //PageName = name of your page
```

b. Navigate sending extra data

Option 1: add a parameter to the constructor

- In the page you want to navigate **to** (eg.: DetailPage), change the constructor asking one or more parameters (of any type):

```
public FooPage(Foo foo)
{
        …
}
```

- Pass the values in the constructor while navigating:

```
        Navigation.PushAsync(new PageName("hello"));
```

Option 2: use a public property

- In the page you want to navigate **to** (eg.: DetailPage), **create a public property** of the **type** of information you want to pass. You can create as many properties as you wish.

```
Eg.:    public string Foo { get; set; }
```

- Set these properties before navigating:

```
        PageName page = new PageName();
        page.Foo = "hello";
        Navigation.PushAsync(page);
```

Other options

(These options require knowledge of databinding / MVVM, and will not be handled in this course. However, you can find them in, amongst others, the eBook.)