Agenda

1. Why use XAML?
2. How to create a XAML Page
3. Adding Behavior

Xamarin University

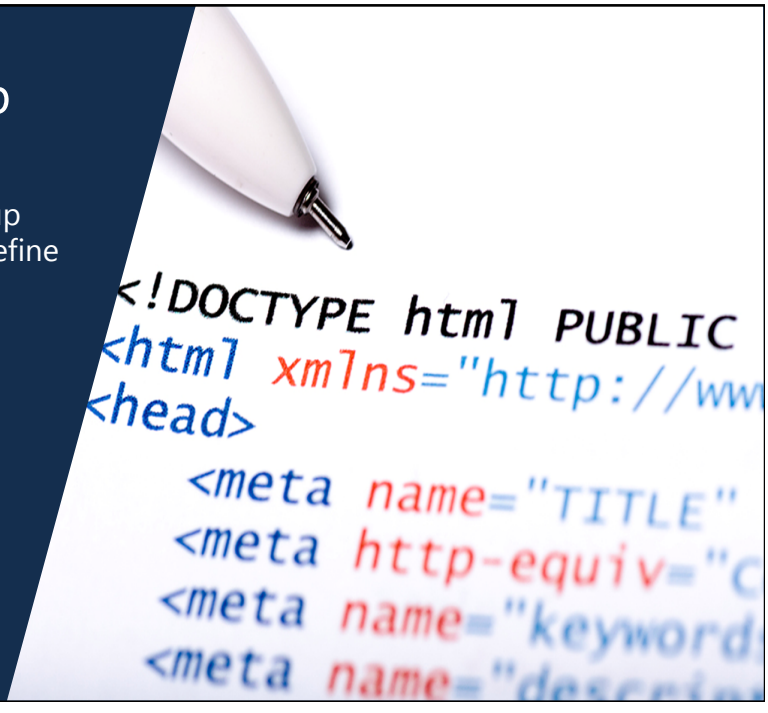
Why use XAML?

Xamarin University

Working in Markup

HTML has taught us that markup languages are a great way to define user interfaces

- toolable
- human readable
- extensible



Xamarin University

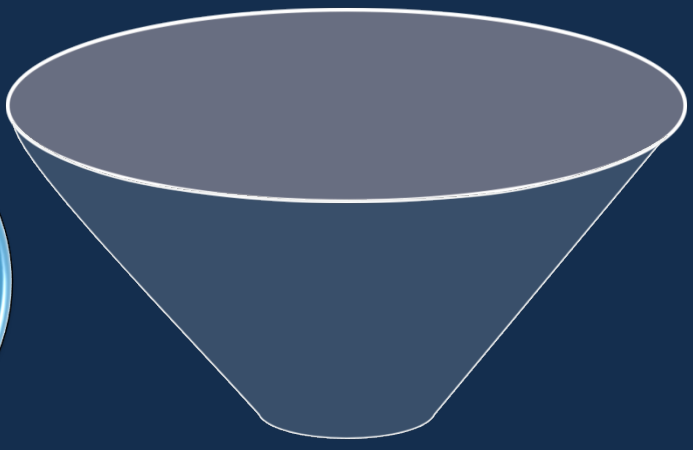
Extensible Application Markup Language (XAML)

XAML

Xamarin Forms + XAML = Sweetness!

Benefits

Separation of UI from Behavior



XAML

Xamarin University

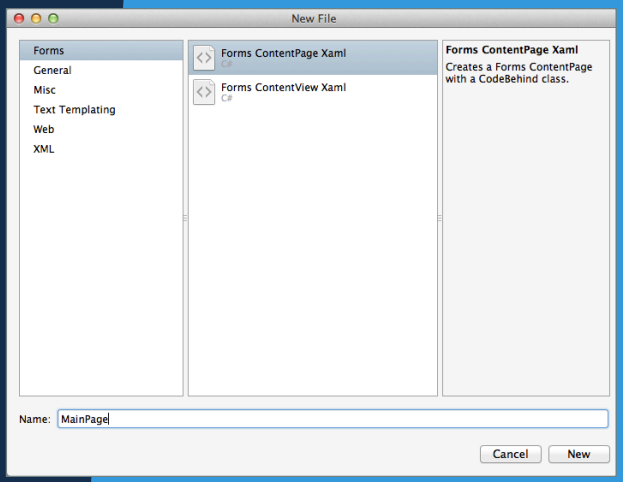
Agenda

1. ~~Why use XAML?~~
2. How to create a XAML Page
3. Adding Behavior

Xamarin University

Creating Pages in XAML

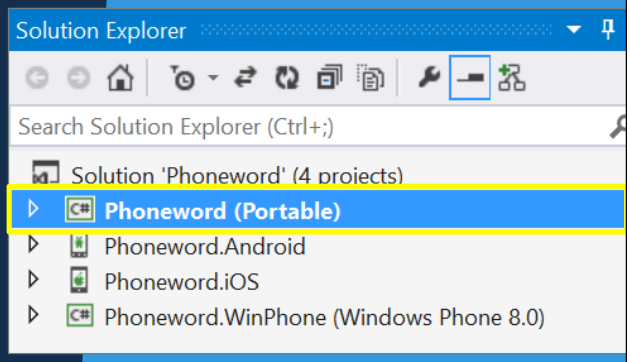
Templates in Visual Studio and Xamarin Studio are used to add XAML content and come in two forms



Xamarin University

Where does the XAML go?

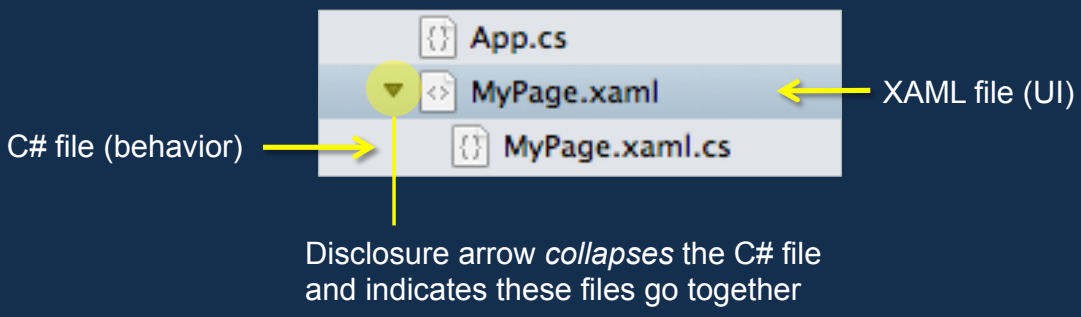
You should always add the XAML content to the *platform-independent* part of your application – this is shared UI and code for all your target platforms!



Xamarin University

What gets created?

- XAML pages have two related files which work together to define the class



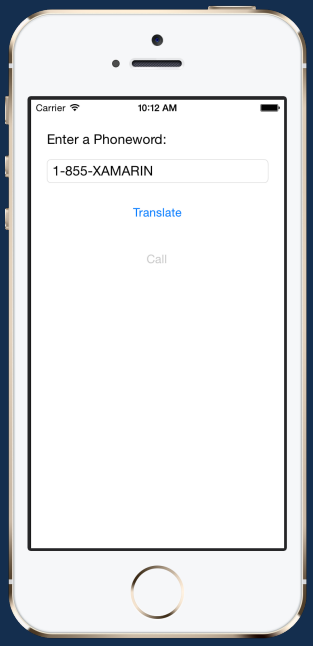
Xamarin University

Let's create a UI

Our goal is to build the UI for a Phone Translation application with a:

- Label (Enter a Phoneword:)
- Entry (1-8555-XAMARIN)
- Button (Translate)
- Button (Call)

Xamarin University



Describing a screen in XAML

XML based: case sensitive, open tags must be closed, etc.

```
<?xml version="1.0" encoding="UTF-8" ?>
<ContentPage ...>
   <StackLayout Padding="20" Spacing="10">
      <Label Text="Enter a Phoneword:" />
      <Entry Placeholder="Number" />
      <Button Text="Translate" />
      <Button Text="Call" IsEnabled="False" />
   </StackLayout>
</ContentPage>
```

Element tags create objects

Child nodes are used to establish relationships

Attributes set properties

C# file contains the behavior for the screen

```
public partial class MyPage : ContentPage
{
    public MyPage ()
    {
        InitializeComponent ();
    }
}
```

Constructor loads XML and creates UI from XAML

Xamarin University

Demo

Xamarin University

Agenda

1. ~~Why use XAML?~~
2. ~~How to create a XAML Page~~
3. Adding Behavior

Xamarin University

Naming elements

Use `x:Name` to associate a field to an element in your code behind file

```
<Entry x:Name="PhoneNumber" Placeholder="Number" />
```

.xaml

```
public partial class MainPage : ContentPage
{
    private Entry PhoneNumber;

    ...
}
```

.cs

field is created for you and is hidden away in another file

Event Handling in XAML

Can wire up to UI events in XAML, handler must be in the code-behind file

```
<Button Content="Translate" ...
        Clicked="OnTranslateClicked" />
```

.xaml

```
void OnTranslateClicked(object sender, EventArgs e)
{
    Button button = (Button)sender;
    ...
}
```

.cs

handler is *not* created for you .. must add this yourself!

Demo

Xamarin University

Summary

1. Why use XAML?
2. How to create a XAML Page
3. Adding Behavior

Xamarin University

Questions?

Xamarin University