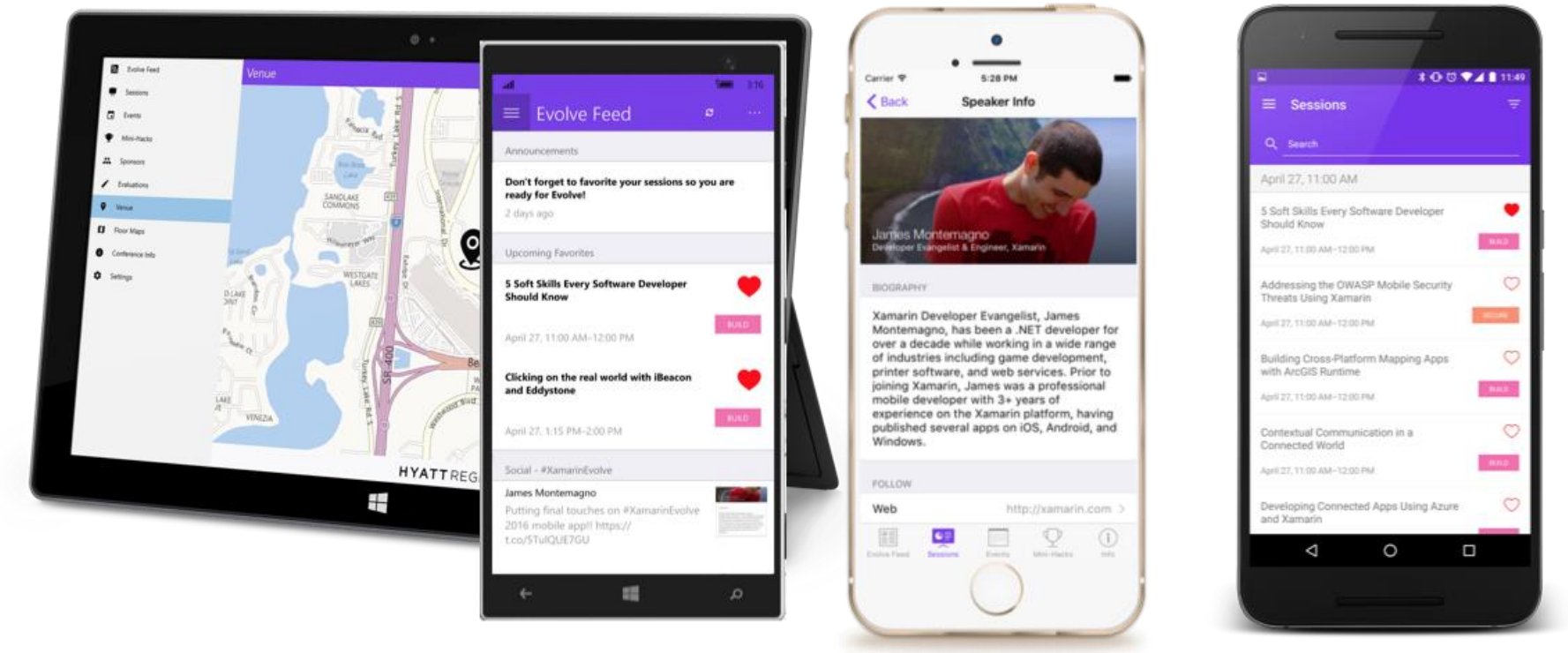
Dev Days

#XamarinDevDays

Cross-Platform Native User Interfaces with Xamarin.Forms

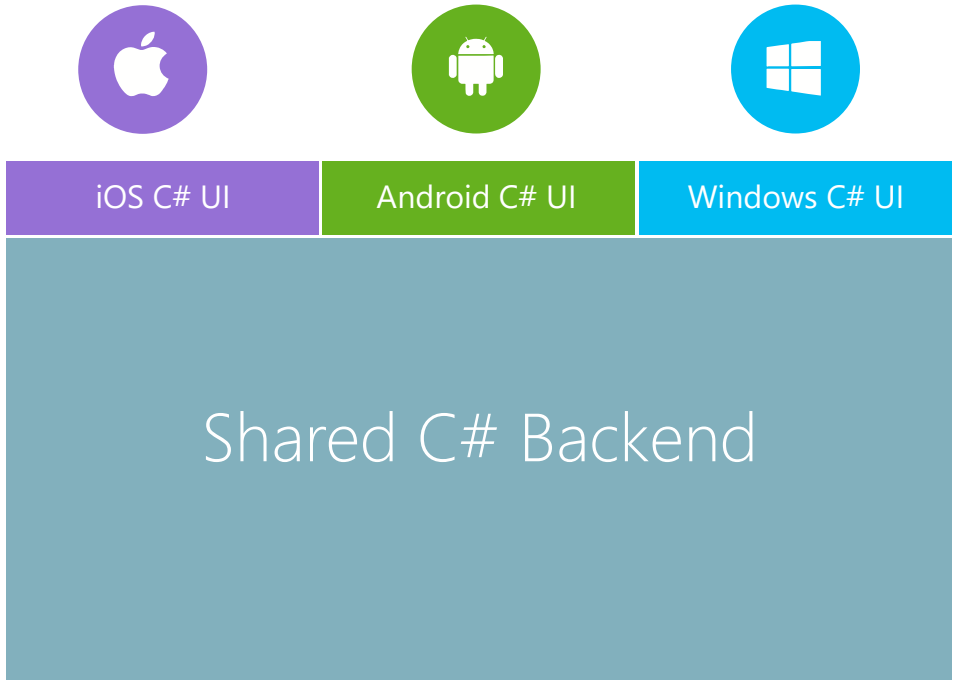
Presenter Name
Presenter Twitter
Presenter Title

Meet Xamarin.Forms

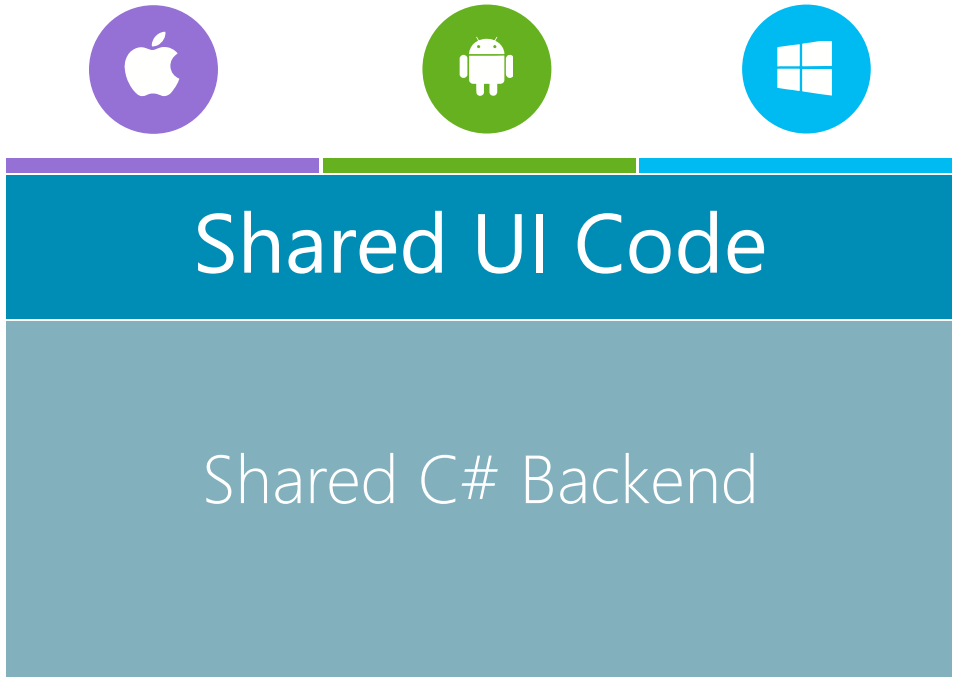


Build native UIs for iOS, Android, and Windows from a single, shared C# codebase.

Xamarin + Xamarin.Forms

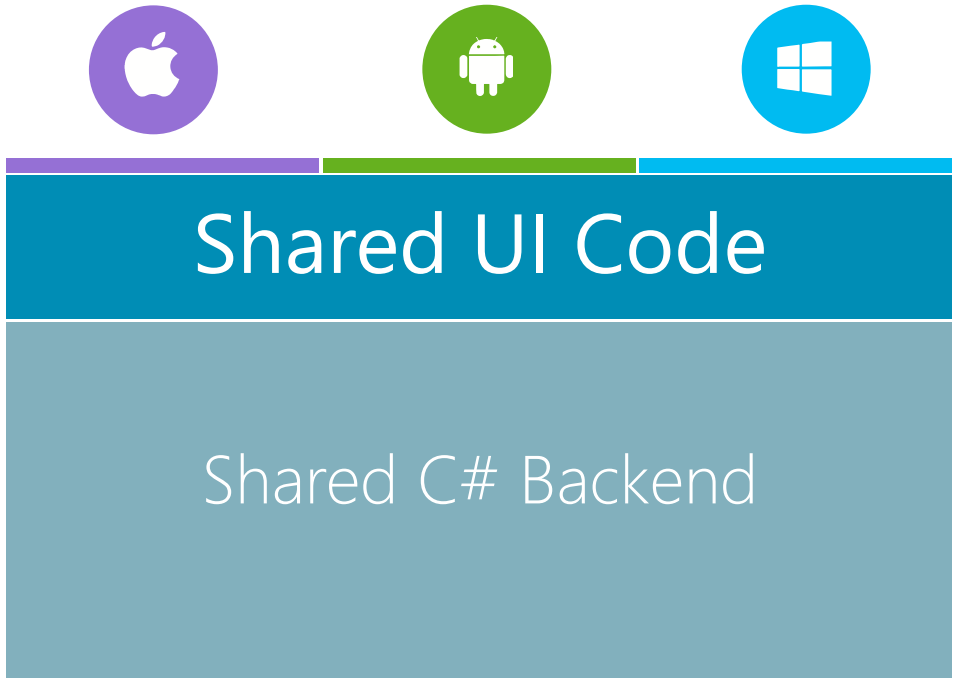


Traditional Xamarin Approach



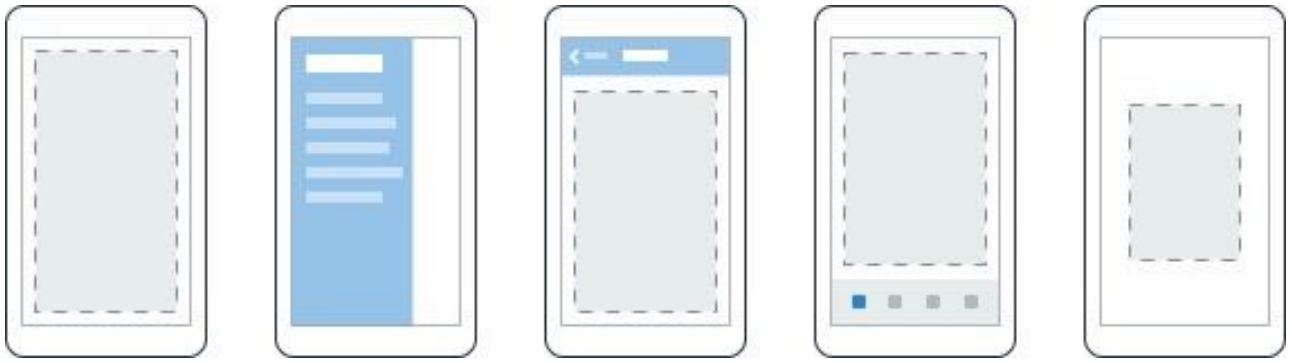
With Xamarin.Forms:
More code-sharing, all native

What's included



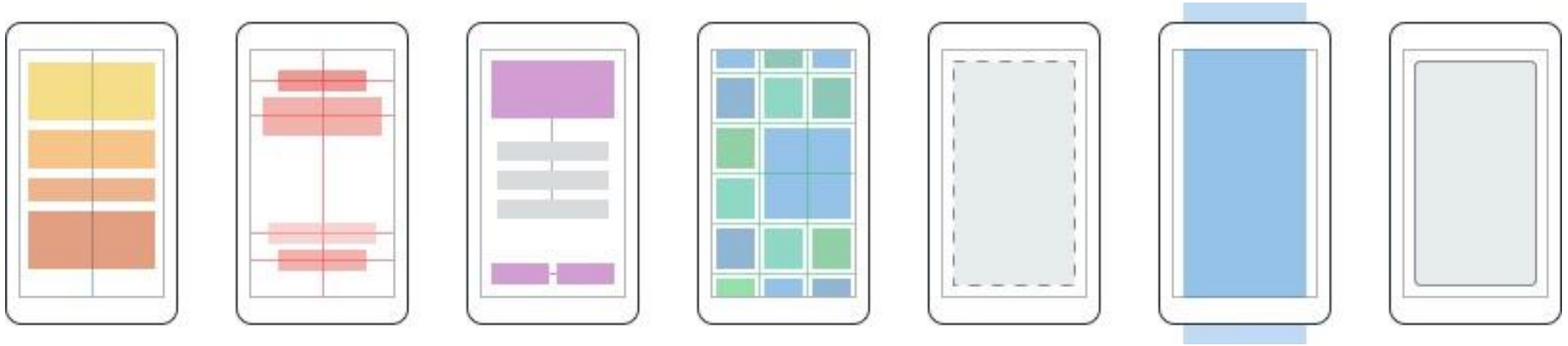
- ✓ 40+ Pages, layouts, and controls
  (Build from code behind or XAML)
- ✓ Two-way data binding
- ✓ Navigation
- ✓ Animation API
- ✓ Dependency Service
- ✓ Messaging Center

Pages



Content MasterDetail Navigation Tabbed Carousel

Layouts



Stack Absolute Relative Grid ContentView ScrollView Frame

Controls

ActivityIndicator	BoxView	Button	DatePicker	Editor
Entry	Image	Label	ListView	Map
OpenGLView	Picker	ProgressBar	SearchBar	Slider
Stepper	TableView	TimePicker	WebView	EntryCell
ImageCell	SwitchCell	TextCell	ViewCell	

Xamarin.Forms Ecosystem

Control Comparison

Windows	Xamarin.Forms
StackPanel	StackLayout
TextBox	Entry
ListBox	ListView
CheckBox	Switch
ProgressBar	ActivityIndicator
Grid	Grid
Label	Label
Button	Button
Image	Image
Date/TimePicker	Date/TimePicker

Binding Comparison

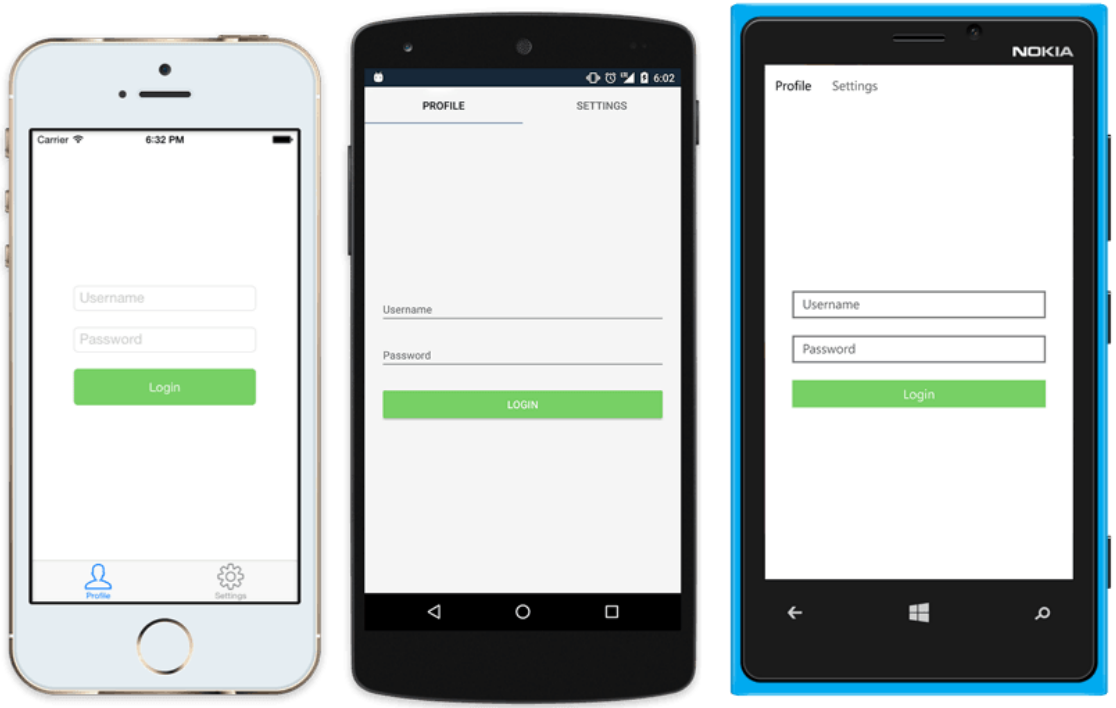
Windows	Xamarin.Forms
DataContext	BindingContext
{Binding Property}	{Binding Property}
ItemsSource	ItemsSource
ItemTemplate	ItemTemplate
DataTemplate	DataTemplate

```
<Label Text="{Binding Color.R,
                      Converter={StaticResource intConverter},
                      ConverterParameter=255,
                      StringFormat='R={0:X2}'}" />
```

```
<ContentPage.Padding>
  <OnPlatform x:TypeArguments="Thickness">
    <OnPlatform.iOS>
      0, 20, 0, 0
    </OnPlatform.iOS>
    <OnPlatform.Android>
      0, 0, 0, 0
    </OnPlatform.Android>
    <OnPlatform.WinPhone>
      0, 0, 0, 0
    </OnPlatform.WinPhone>
  </OnPlatform>
</ContentPage.Padding>
```

Platform Customization

Native UI from shared code




```
<?xml version="1.0" encoding="UTF-8"?>
<TabbedPage xmlns="http://xamarin.com/schemas/2014/forms"
            xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
            x:Class="MyApp.MainPage">
<TabbedPage.Children>
<ContentPage Title="Profile" Icon="Profile.png">
    <StackLayout Spacing="20" Padding="20"
                 VerticalOptions="Center">
        <Entry Placeholder="Username"
               Text="{Binding Username}"/>
        <Entry Placeholder="Password"
               Text="{Binding Password}"
               IsPassword="true"/>
        <Button Text="Login" TextColor="White"
                BackgroundColor="#77D065"
                Command="{Binding LoginCommand}"/>
    </StackLayout>
</ContentPage>
<ContentPage Title="Settings" Icon="Settings.png">
    <!-- Settings -->
</ContentPage>
</TabbedPage.Children>
</TabbedPage>
```

Xamarin.Forms

Demo

What's new?

Xamarin.Forms 2.0!

Xamarin.Forms is Open Source!

open.xamarin.com

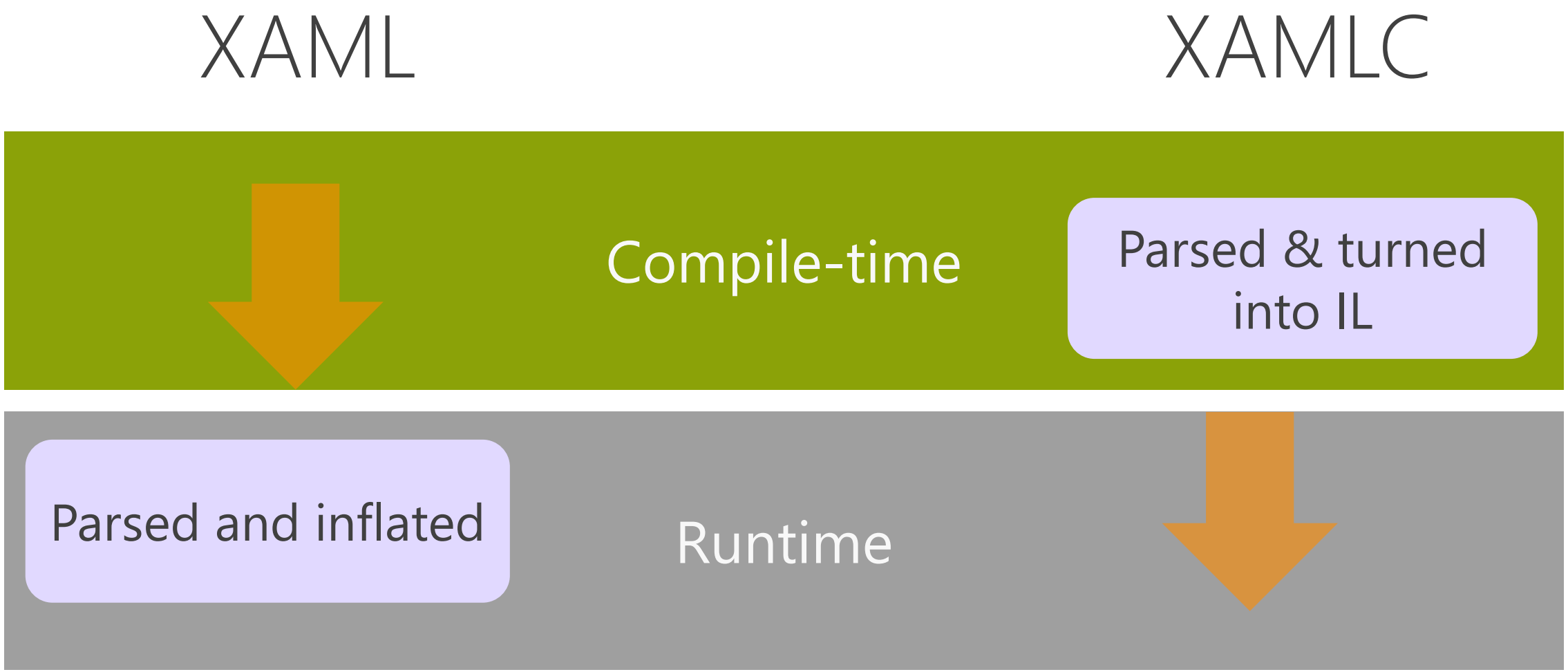
Performance
Performance
Performance

ListView CachingStrategy

```
public enum ListViewCachingStrategy
{
    RetainElement = 0,
    RecycleElement,
}
```

✓ Improves scrolling performance
✓ Old behavior is currently the default (but likely to change)

XAML Compilation

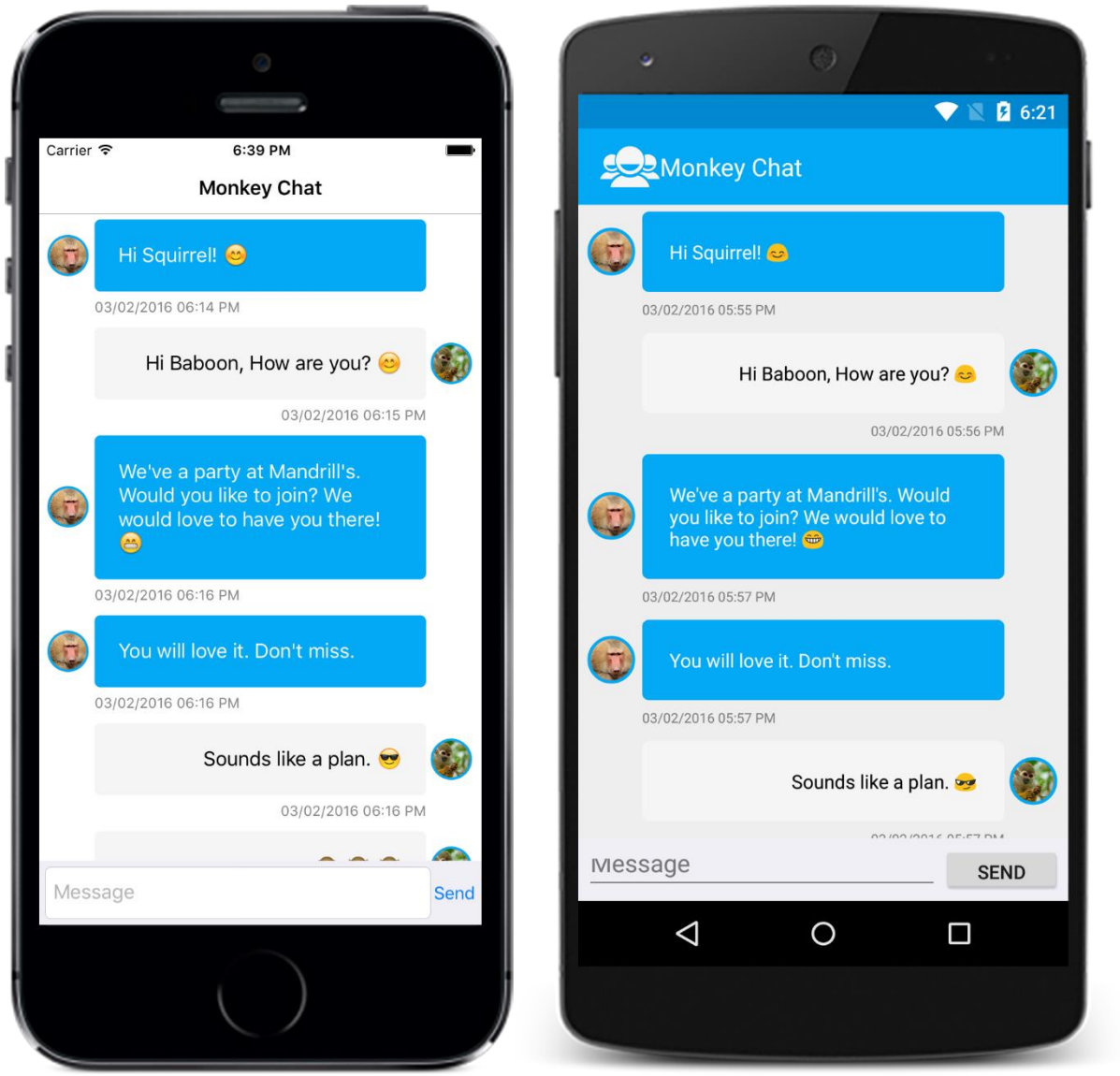


Benefits

- ✓ Faster Loading
- ✓ Smaller App Size
- ✓ See XAML errors at build time

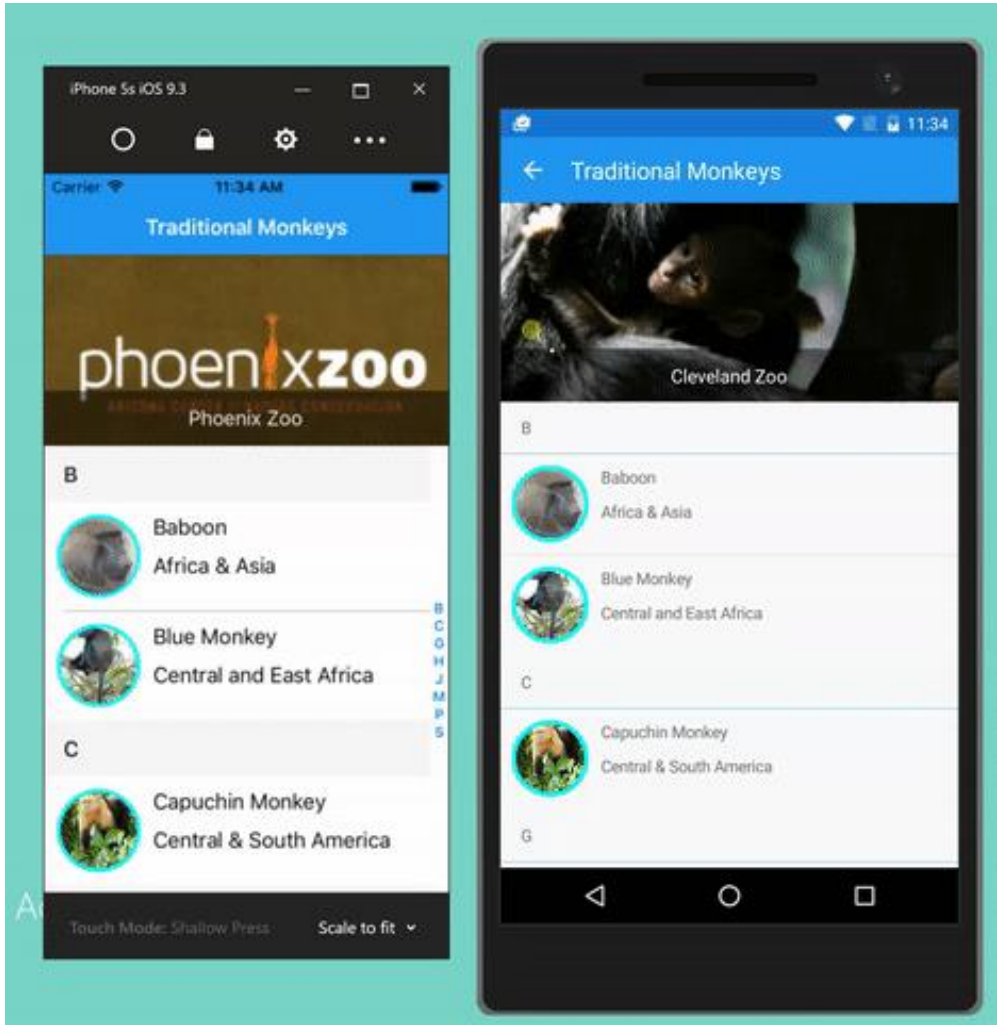
Design & Controls

Data Templates



CarouselView

- Embeddable Carousel
- ItemTemplate & DataTemplates
- Highly customizable & Virtualized



Effects I

✓ Custom renderer “lite”
✓ Change properties on the native control
✓ Optional
✓ “stringly-typed”

X *No methods or events*
X *No replacing the control*

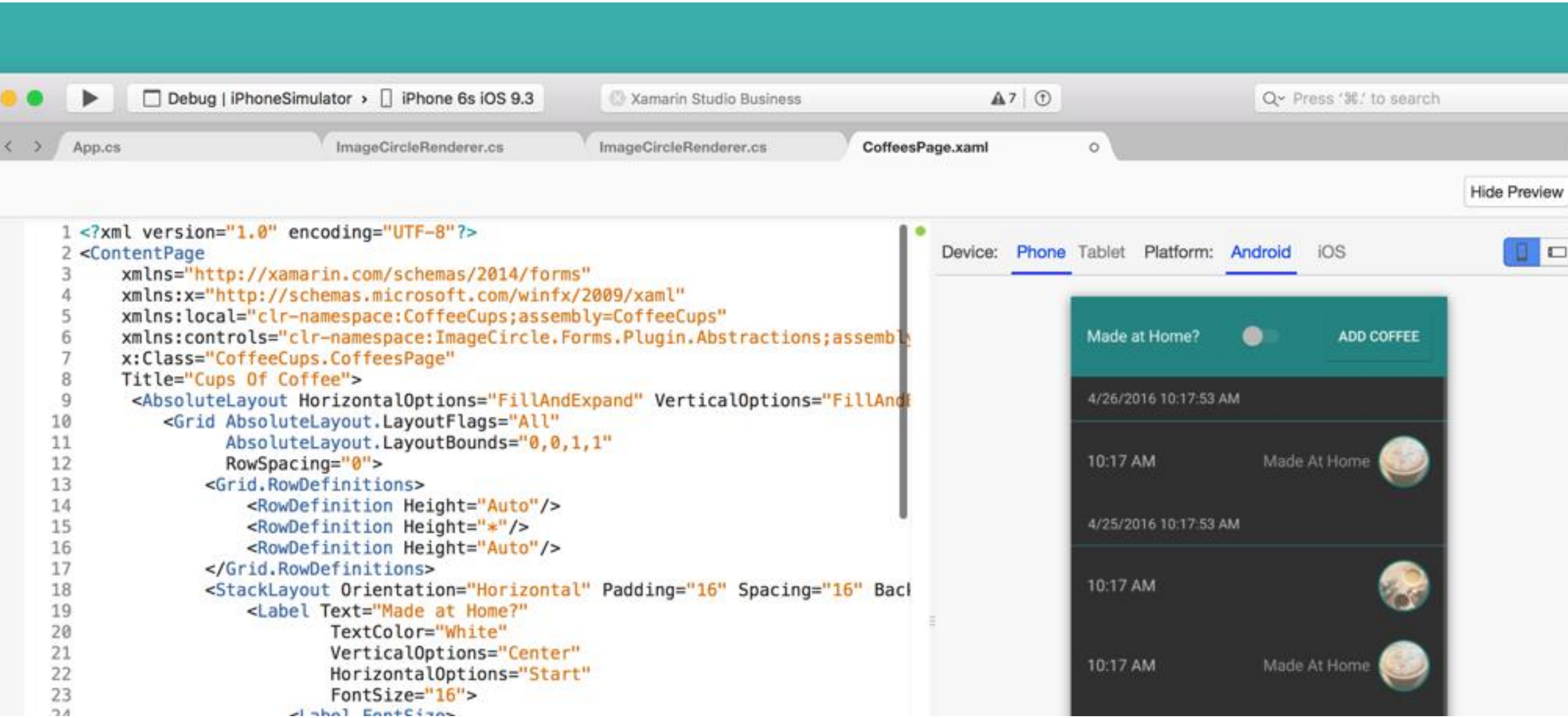
Effects II

```
entry.Effects.Add (Effect.Resolve("Xamarin.BorderEffect"));
```

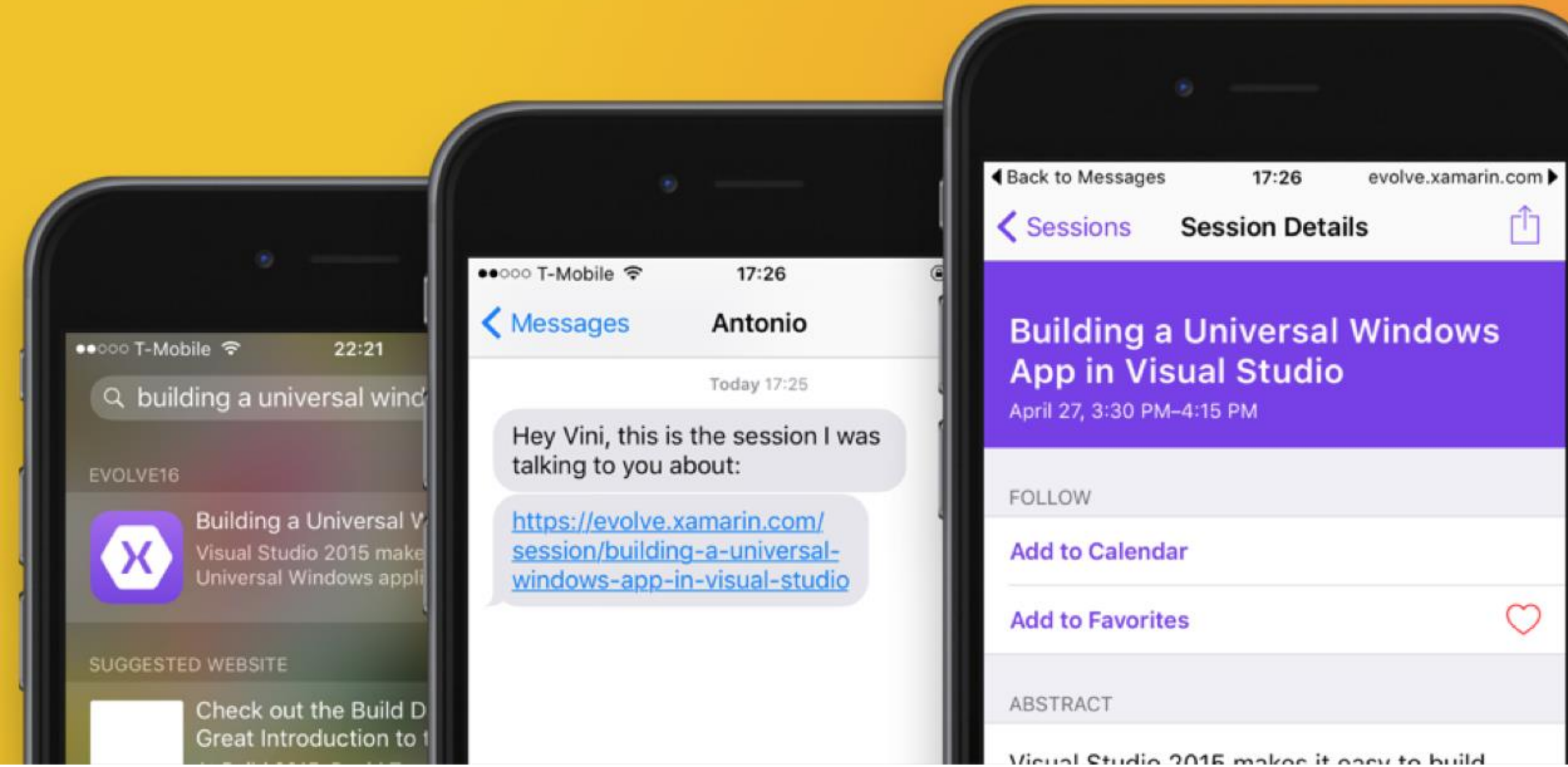
```
// located inside iOS specific codebase
public class BorderEffect : PlatformEffect
{
  protected override void OnAttached ()
  {
    Control.Layer.BorderColor = UIColor.Purple.CGColor;
    Control.Layer.BorderWidth = 2;
  }

  protected override void OnDetached ()
  {
    Control.Layer.BorderWidth = 0;
  }
}
```

So much more

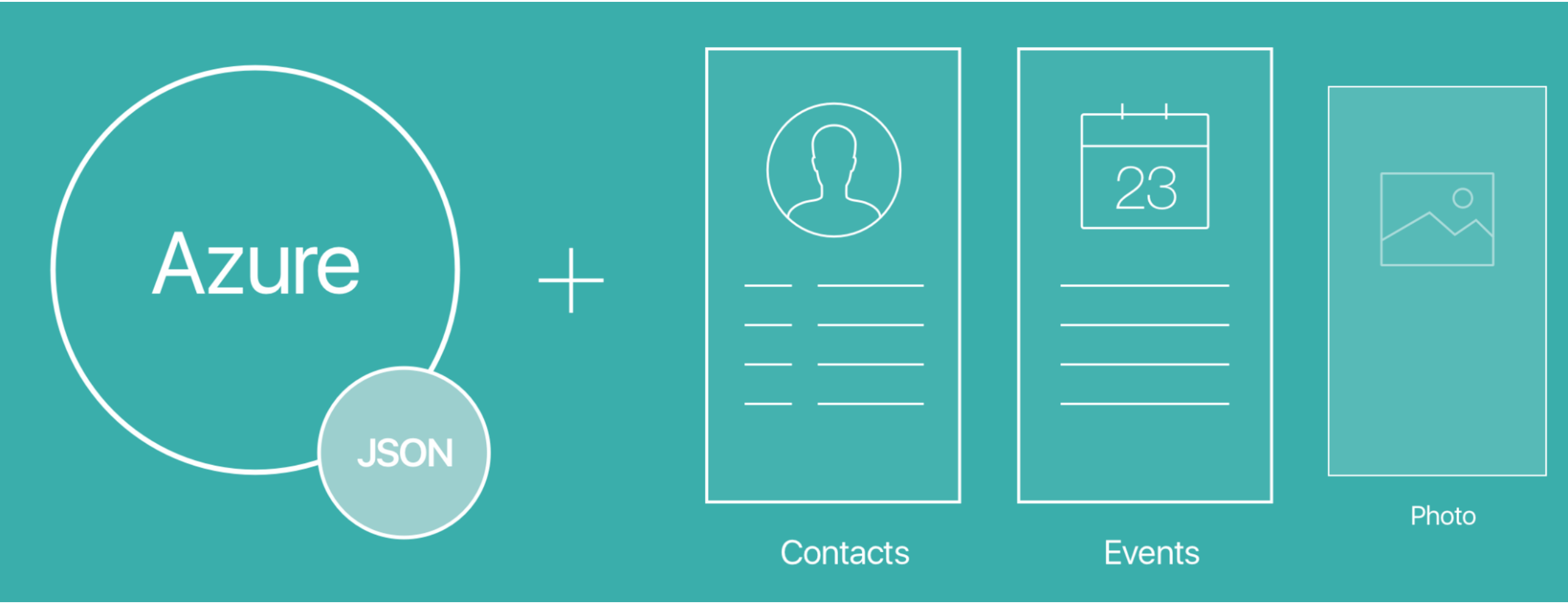
XAML Previewer



URL Navigation

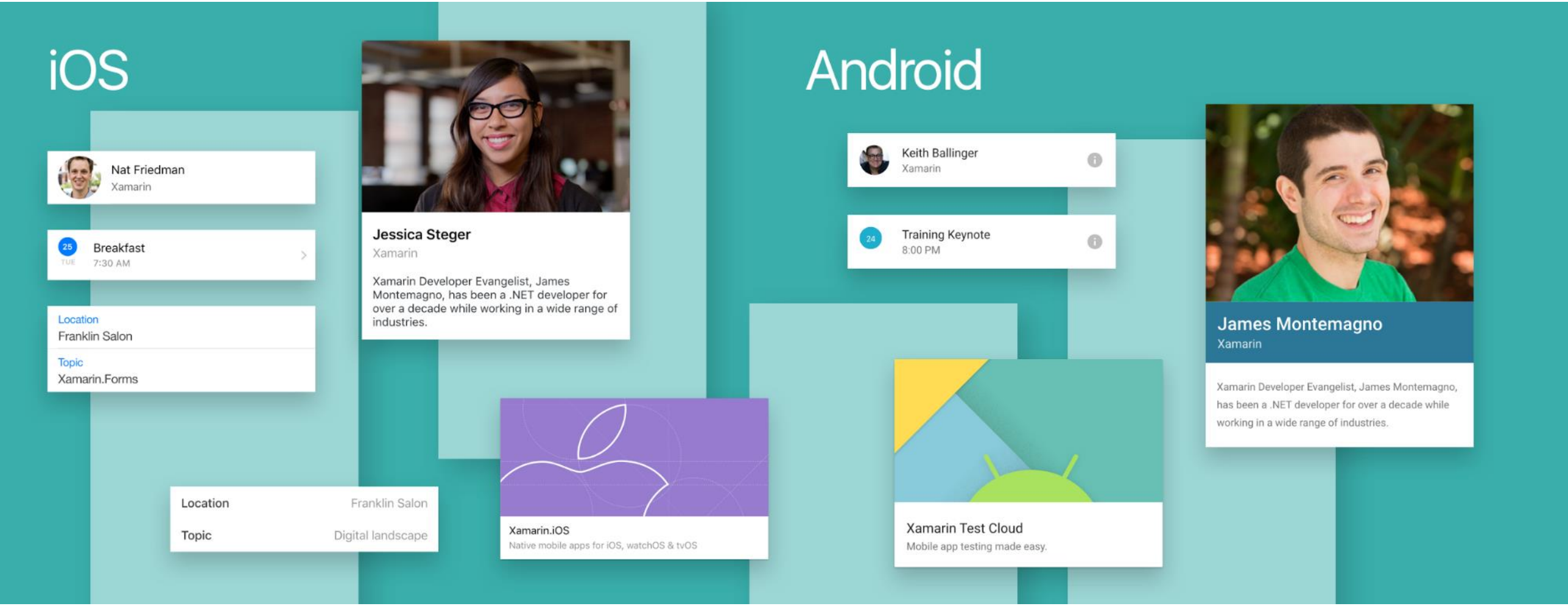


DataPages



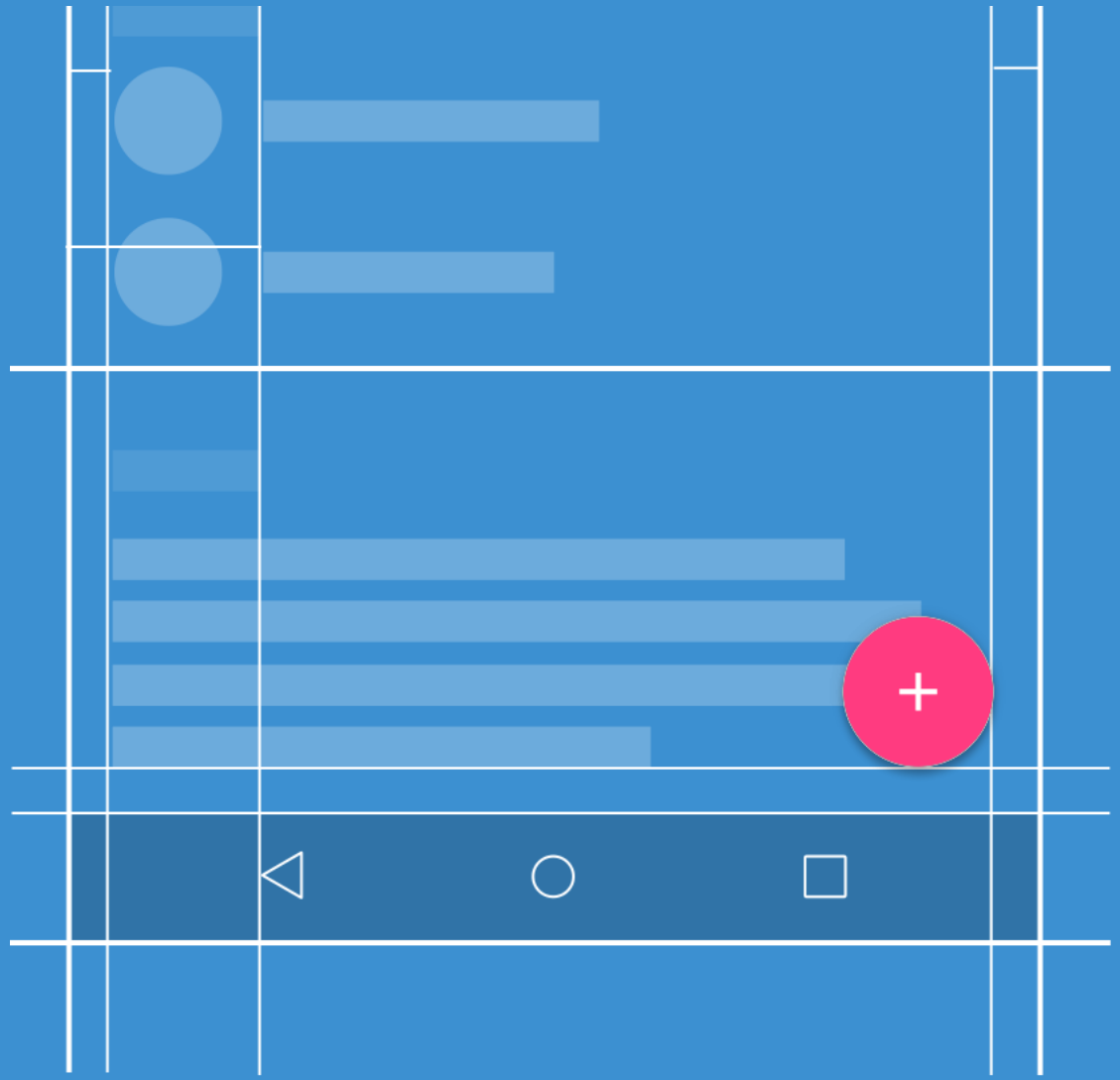
https://developer.xamarin.com/guides/xamarin-forms/datapages/

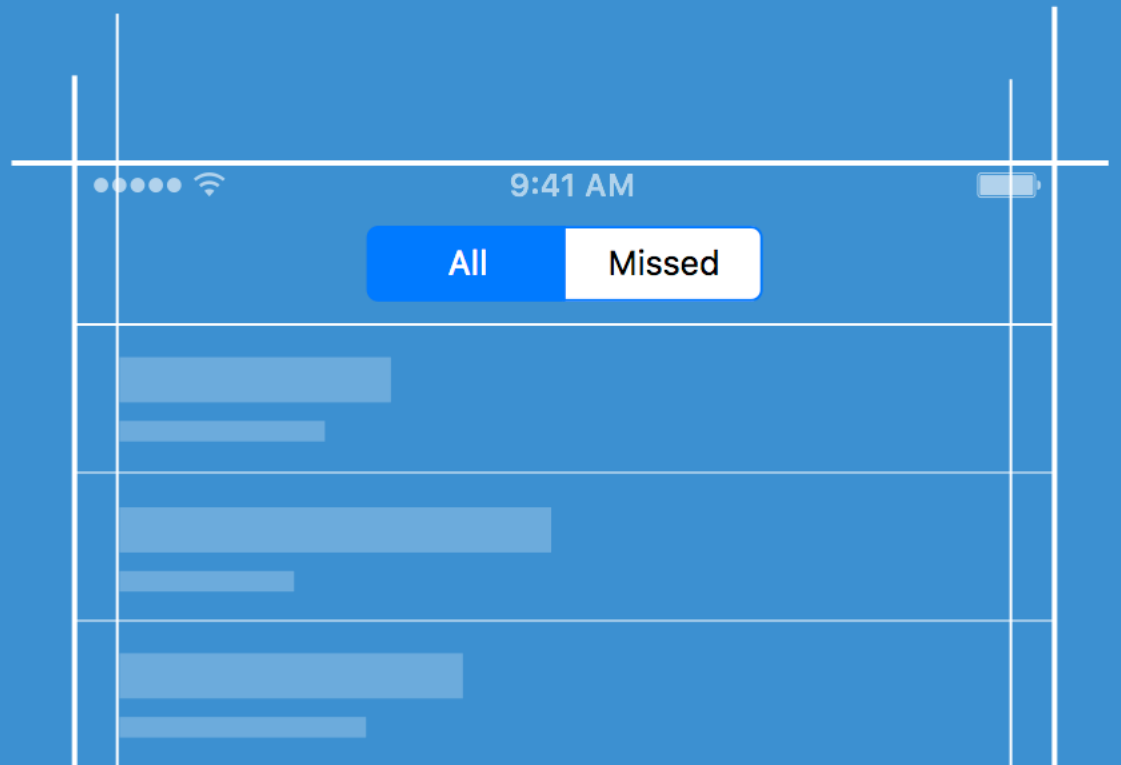
Themes



https://developer.xamarin.com/guides/xamarin-forms/themes/

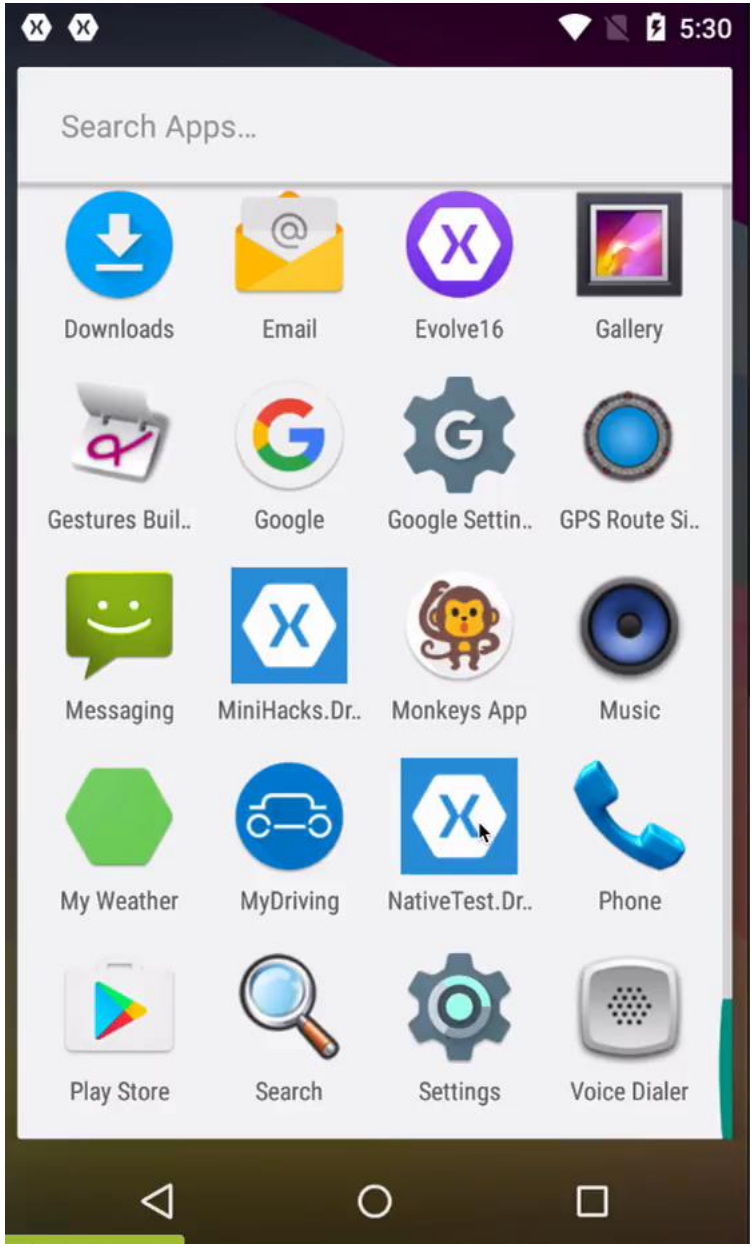
Native Embedding





Native Embedding

```
var stack = new StackLayout
{
    Padding = 50,
    HorizontalOptions = LayoutOptions.Center,
};

#if __ANDROID__
var fab = new CheckableFab(Forms.Context);
fab.SetImageResource(Droid.Resource.Drawable.ic_fancy_fab_icon);
fab.Click += async (sender, e) =>
{
    await MainPage.DisplayAlert("Native FAB Clicked",
                                "Whoa!!!!!!", "OK");
};
stack.Children.Add(fab);
#endif
```



20 Minute Break

Presenter First Name
Presenter Last Name
Presenter Title

Presenter e-mail

blogs

Twitter