

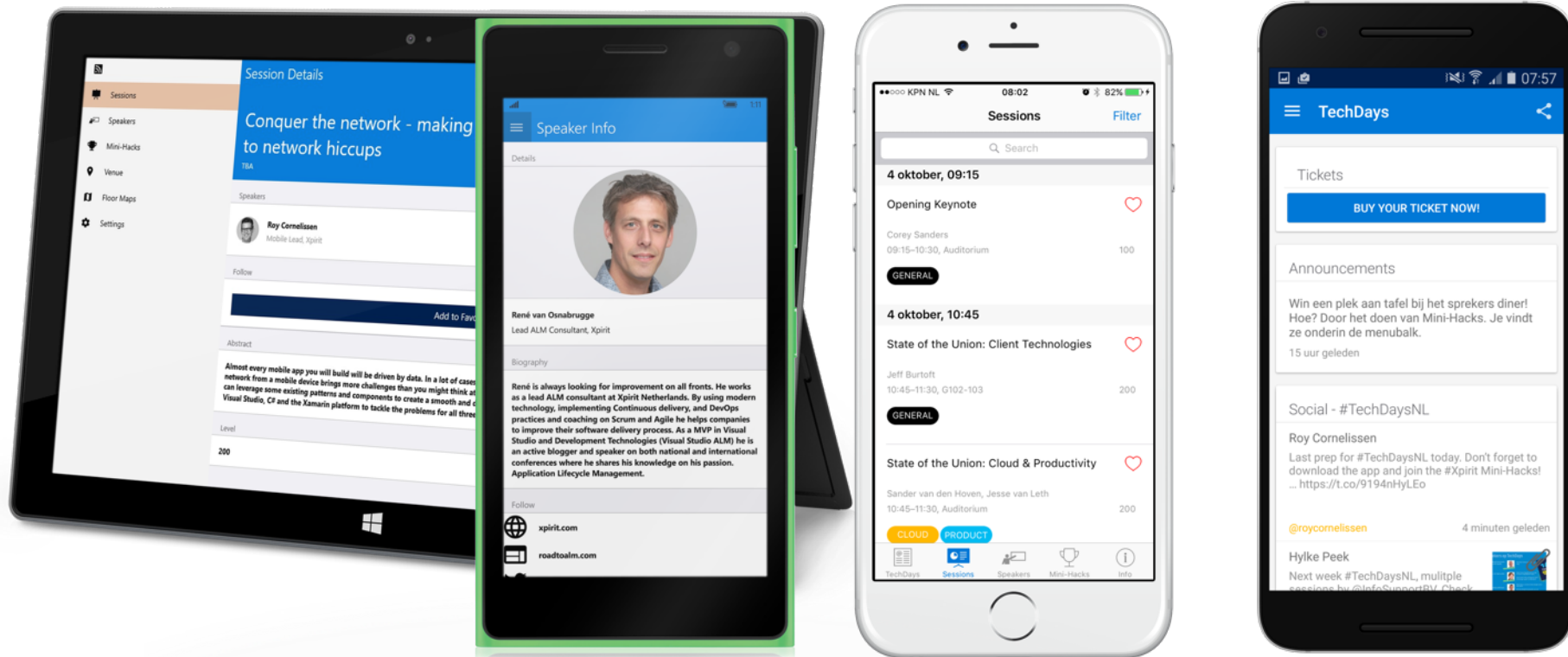


# Kickstarter

# Cross-Platform Native User Interfaces with Xamarin.Forms

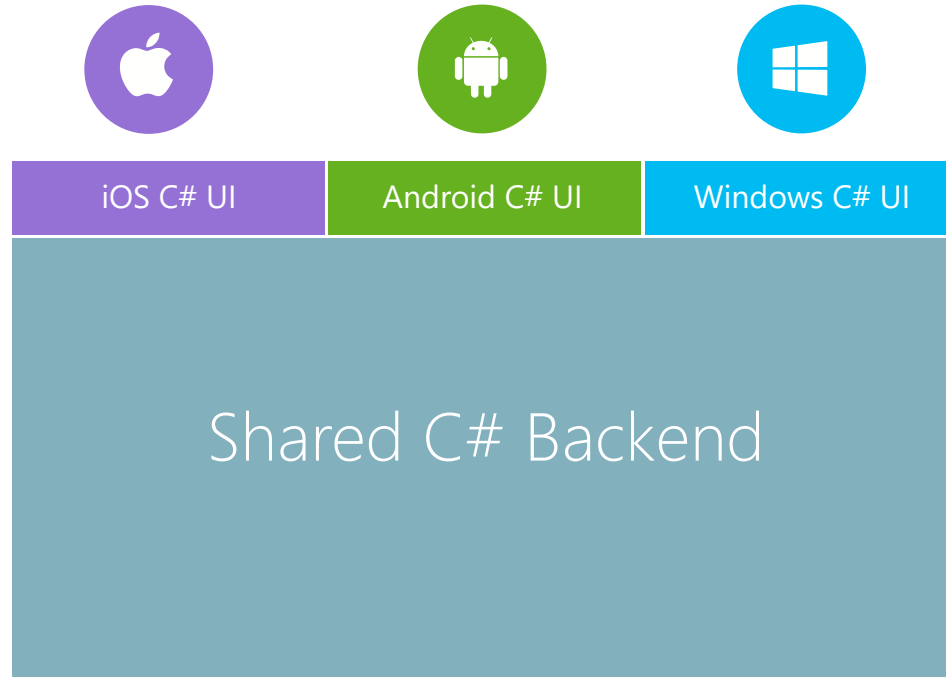
Roy Cornelissen

# Meet Xamarin.Forms

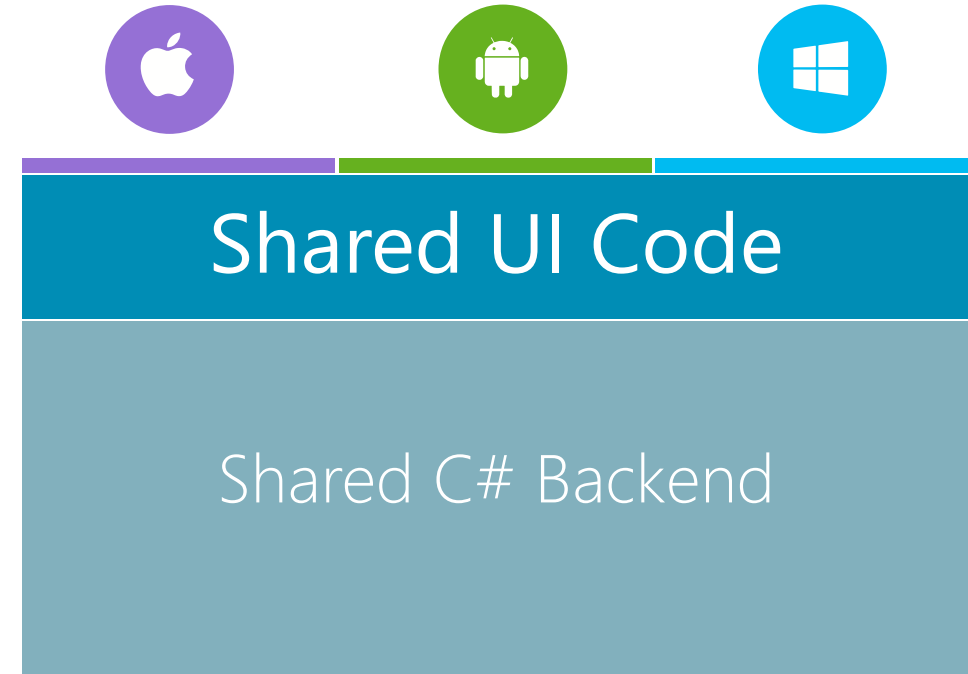


Build native UIs for iOS, Android, and Windows  
from a single, shared C# codebase.

# Xamarin + Xamarin.Forms

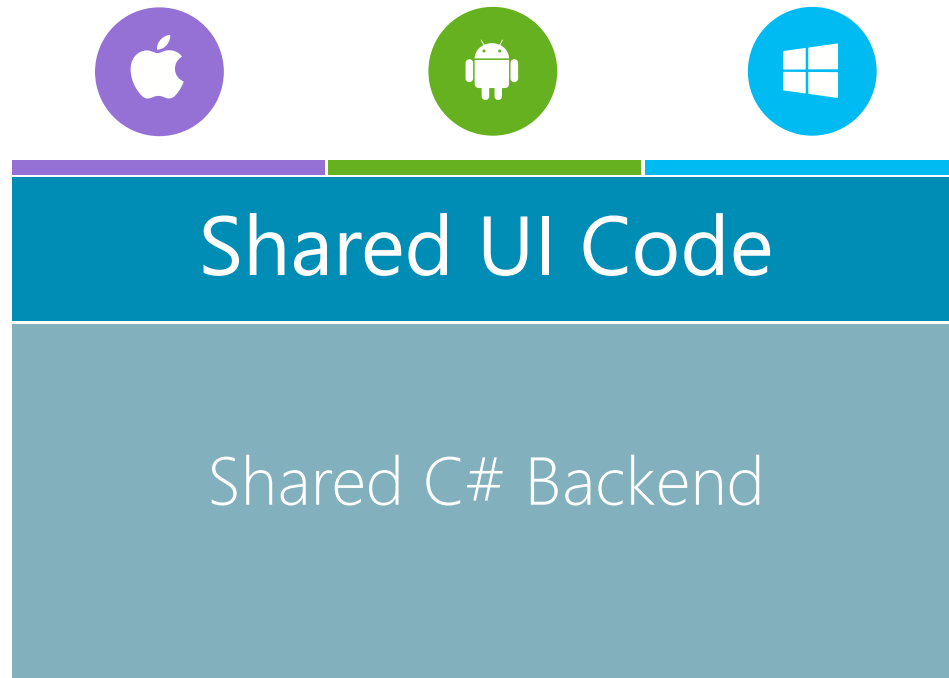


Traditional Xamarin  
Approach



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

# What's included

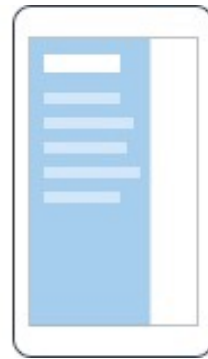


- ✓ 40+ Pages, layouts, and controls  
(Build from code behind or XAML)
- ✓ Two-way data binding
- ✓ Navigation
- ✓ Animation API
- ✓ Dependency Service
- ✓ Messaging Center

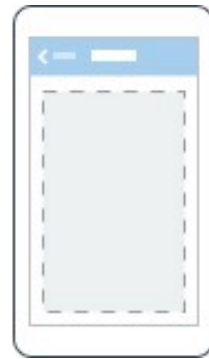
# Pages



Content



MasterDetail

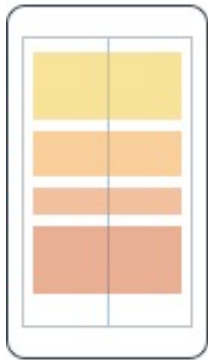


Navigation

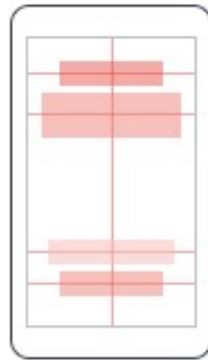


Tabbed

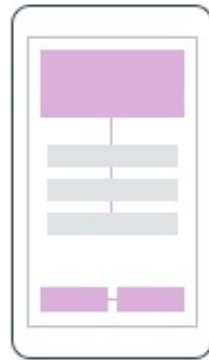
# Layouts



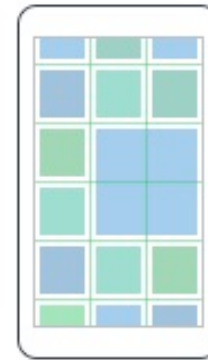
Stack



Absolute



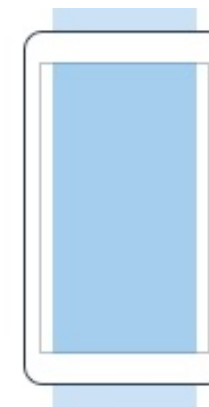
Relative



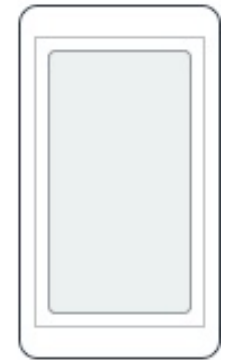
Grid



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





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

# Control Comparison

Windows	Xamarin.Forms
<b>DataContext</b>	<b>BindingContext</b>
{Binding Property}	{Binding Property}
ItemsSource	ItemsSource
ItemTemplate	ItemTemplate
DataTemplate	DataTemplate

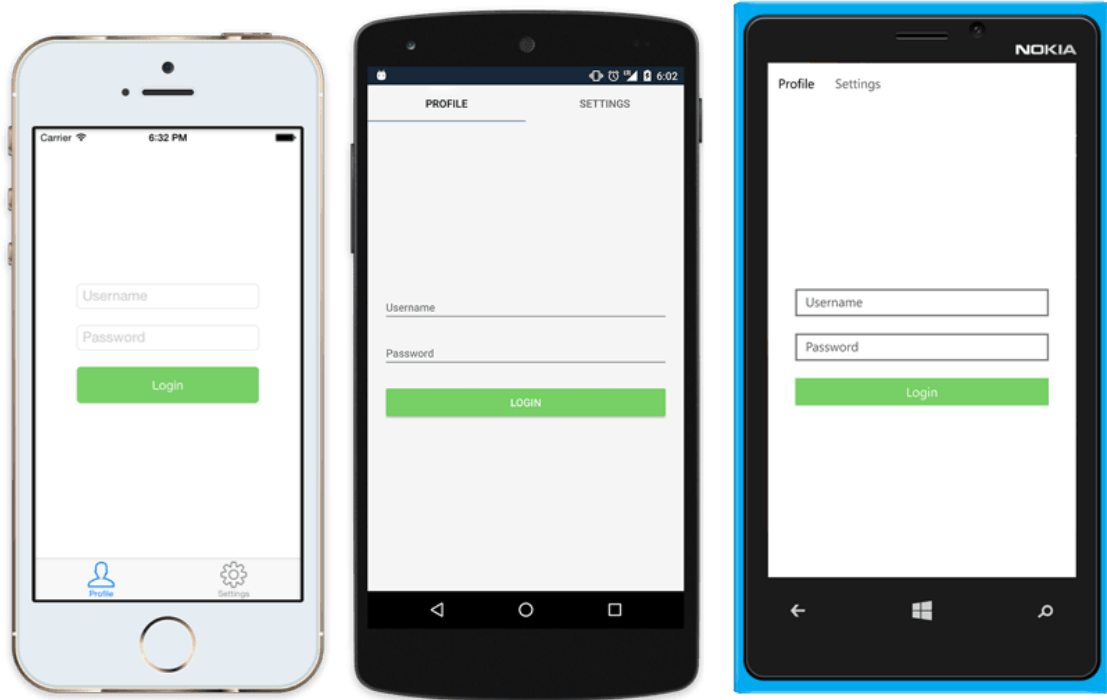
# Binding Comparison

```
<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 2.3!

Xamarin.Forms  
is Open Source!

[open.xamarin.com](https://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

XAML

XAMLC



Compile-time

Parsed & turned  
into IL

Parsed and inflated

Runtime

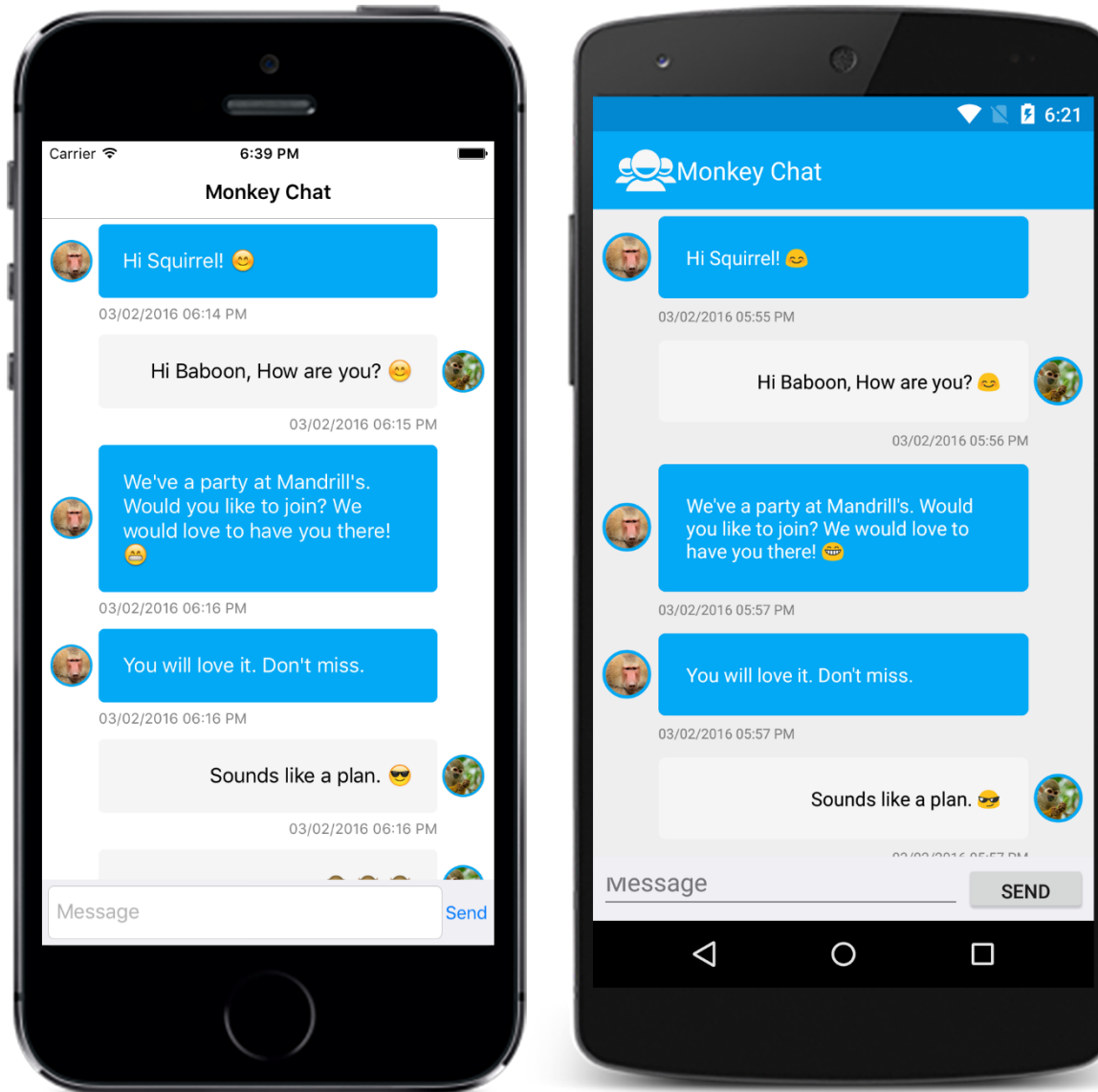


# Benefits

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

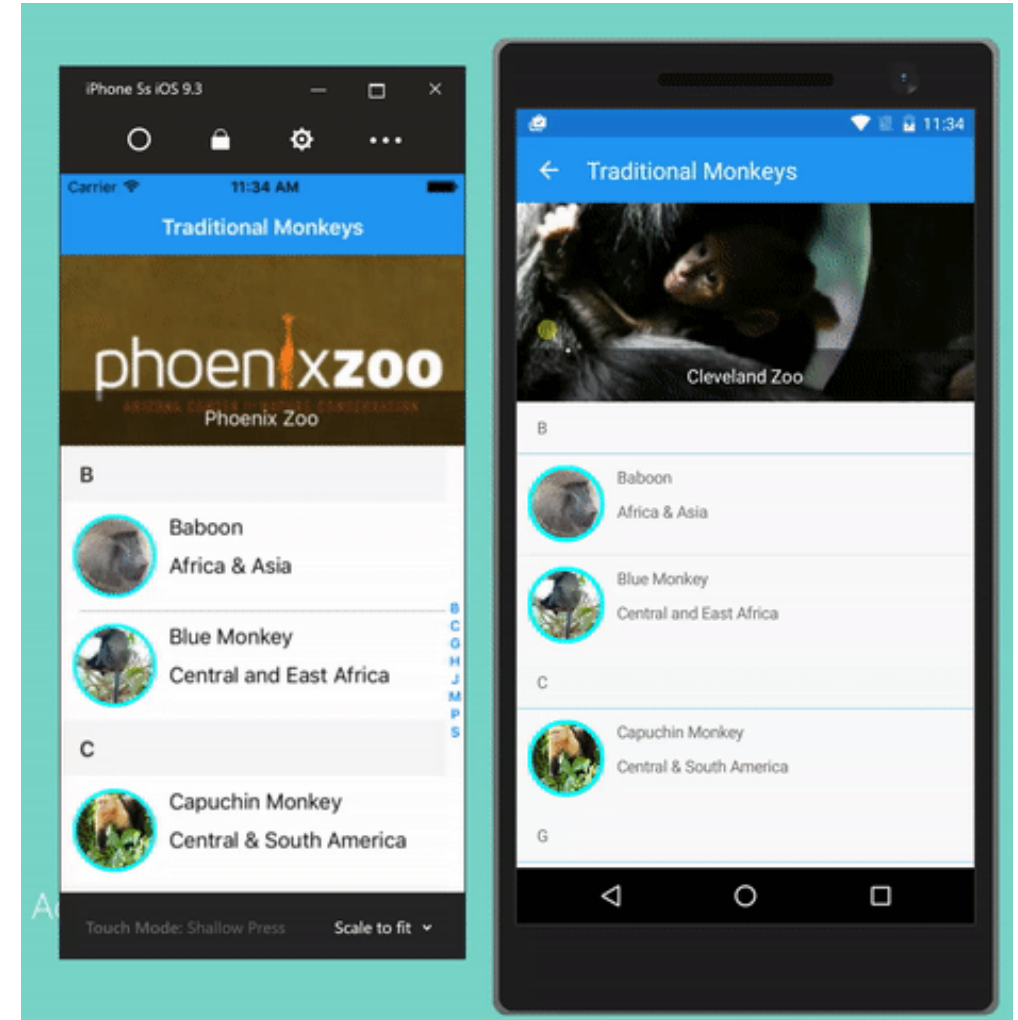
# Design & Controls

# 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

The screenshot shows the Xamarin Studio Business interface. The top bar includes a search bar with the text "Press '⌘.' to search". The file explorer shows the following files: App.cs, ImageCircleRenderer.cs, ImageCircleRenderer.cs, and CoffeesPage.xaml. The XAML code is displayed in the left pane, and the preview of the application is shown in the right pane.

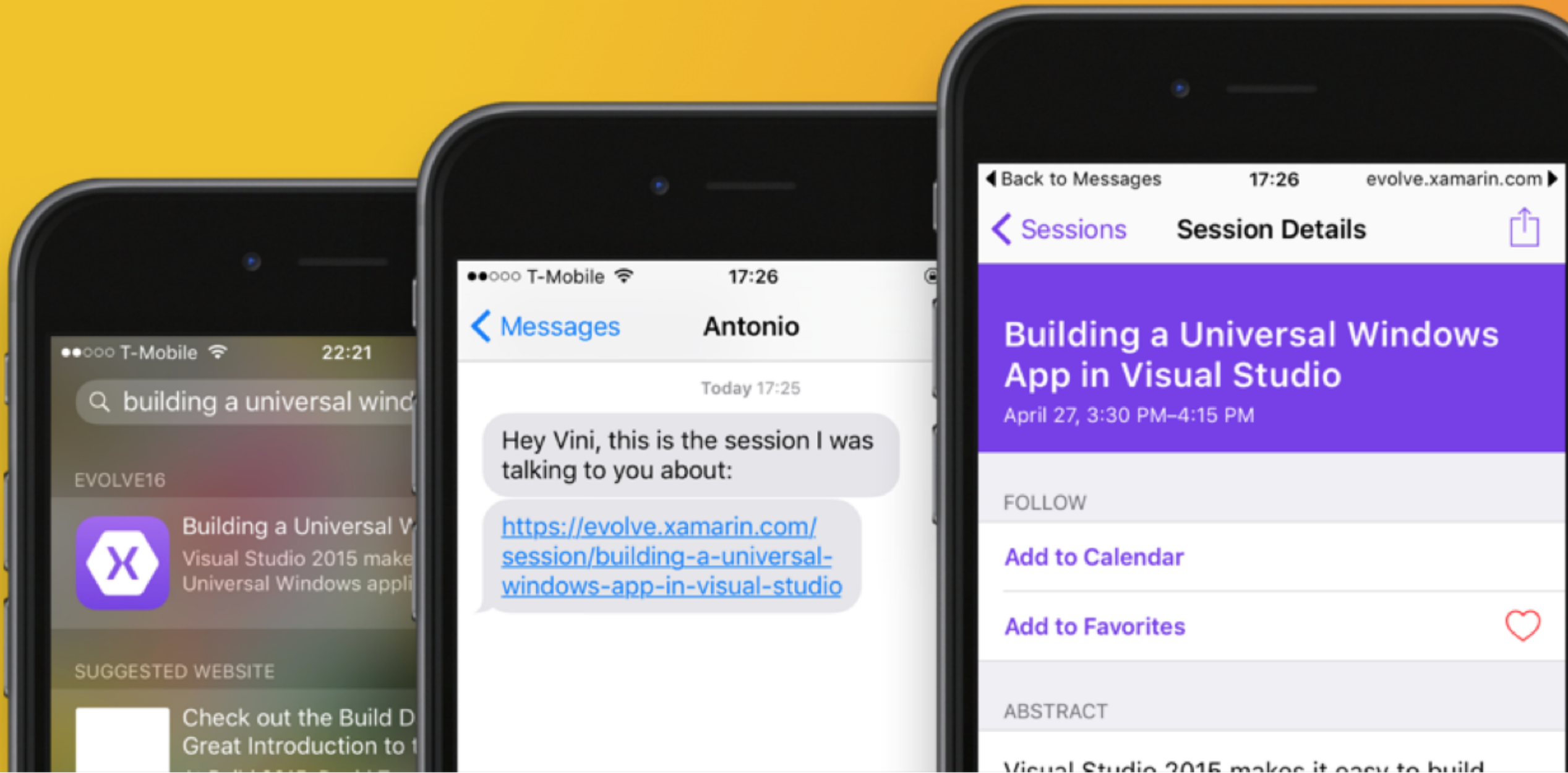
**XAML Code:**

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <ContentPage
3   xmlns="http://xamarin.com/schemas/2014/forms"
4   xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
5   xmlns:local="clr-namespace:CoffeeCups;assembly=CoffeeCups"
6   xmlns:controls="clr-namespace:ImageCircle.Forms.Plugin.Abstractions;assembly=ImageCircle.Forms.Plugin.Abstractions"
7   x:Class="CoffeeCups.CoffeesPage"
8   Title="Cups Of Coffee">
9   <AbsoluteLayout HorizontalOptions="FillAndExpand" VerticalOptions="FillAndExpand">
10     <Grid AbsoluteLayout.LayoutFlags="All"
11       AbsoluteLayout.LayoutBounds="0,0,1,1"
12       RowSpacing="0">
13       <Grid.RowDefinitions>
14         <RowDefinition Height="Auto"/>
15         <RowDefinition Height="*" />
16         <RowDefinition Height="Auto"/>
17       </Grid.RowDefinitions>
18       <StackLayout Orientation="Horizontal" Padding="16" Spacing="16" BackgroundColor="Teal">
19         <Label Text="Made at Home?"
20           TextColor="White"
21           VerticalOptions="Center"
22           HorizontalOptions="Start"
23           FontSize="16">
24           <Label Text="ADD COFFEE" />
25         </StackLayout>
26       </Grid>
27     </AbsoluteLayout>
28   </ContentPage>
```

**Preview:**

The preview shows the application running on an Android phone. The interface includes a toggle switch for "Made at Home?" and a button labeled "ADD COFFEE". Below this, there is a list of coffee entries, each showing a timestamp (e.g., "4/26/2016 10:17:53 AM"), a time (e.g., "10:17 AM"), and a status (e.g., "Made At Home") next to a coffee cup icon.

# URL Navigation



# Themes

## iOS



**Nat Friedman**  
Xamarin

25

**Breakfast**  
TUE 7:30 AM



**Location**

Franklin Salon

**Topic**

Xamarin.Forms

**Location**

Franklin Salon

**Topic**

Digital landscape



**Jessica Steger**

Xamarin

Xamarin Developer Evangelist, James Montemagno, has been a .NET developer for over a decade while working in a wide range of industries.



**Xamarin.iOS**

Native mobile apps for iOS, watchOS & tvOS

## Android



**Keith Ballinger**  
Xamarin



24

**Training Keynote**  
8:00 PM



**Xamarin Test Cloud**

Mobile app testing made easy.



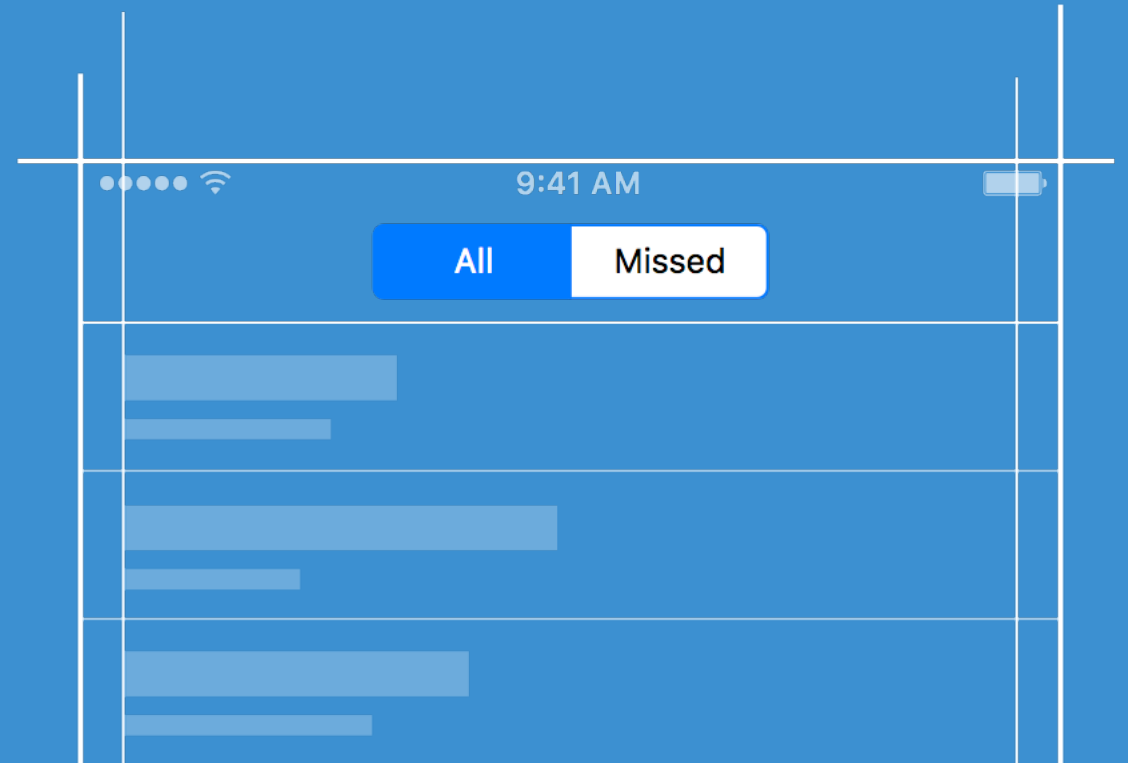
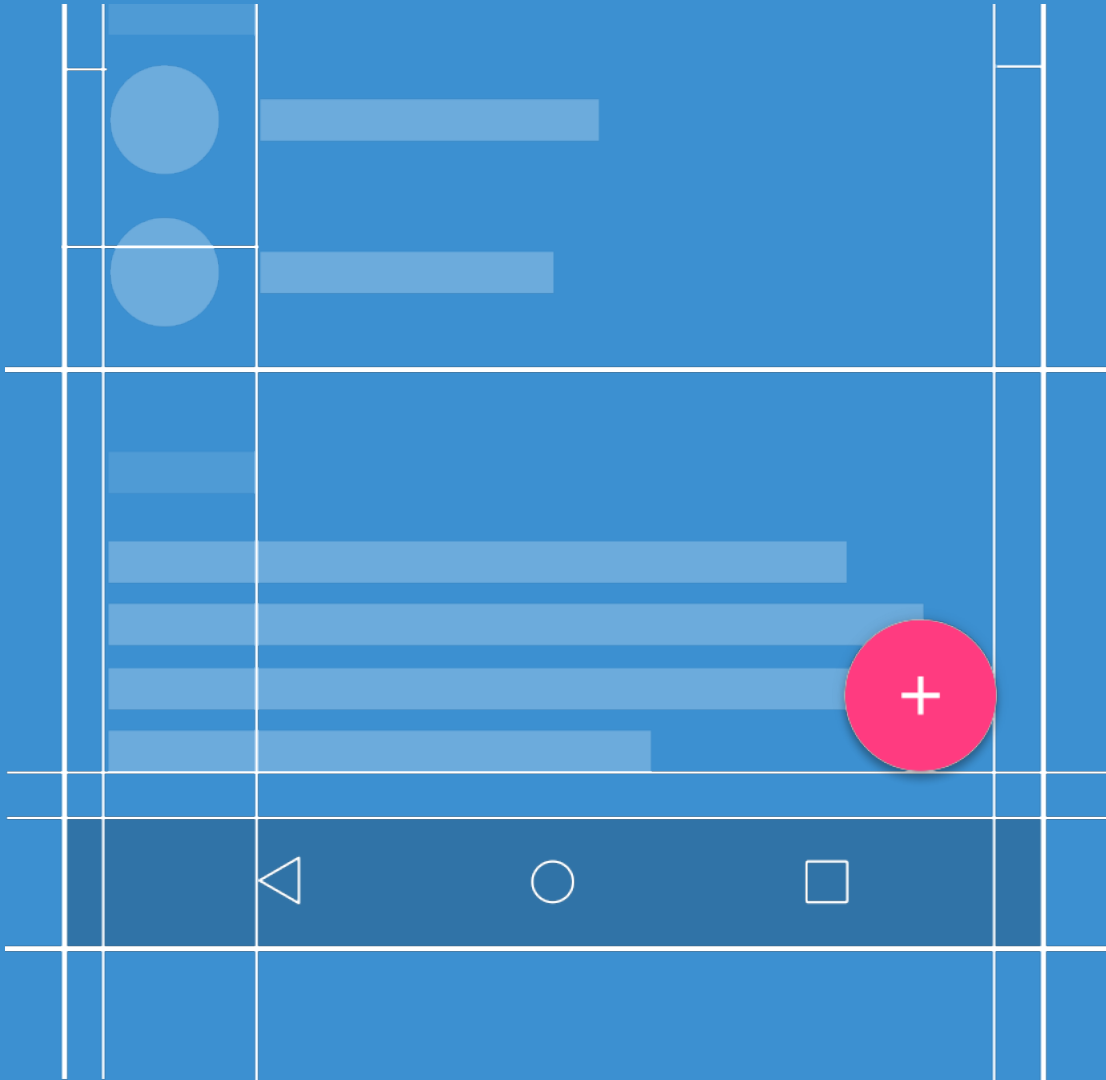
**James Montemagno**

Xamarin

Xamarin Developer Evangelist, James Montemagno, has been a .NET developer for over a decade while working in a wide range of industries.

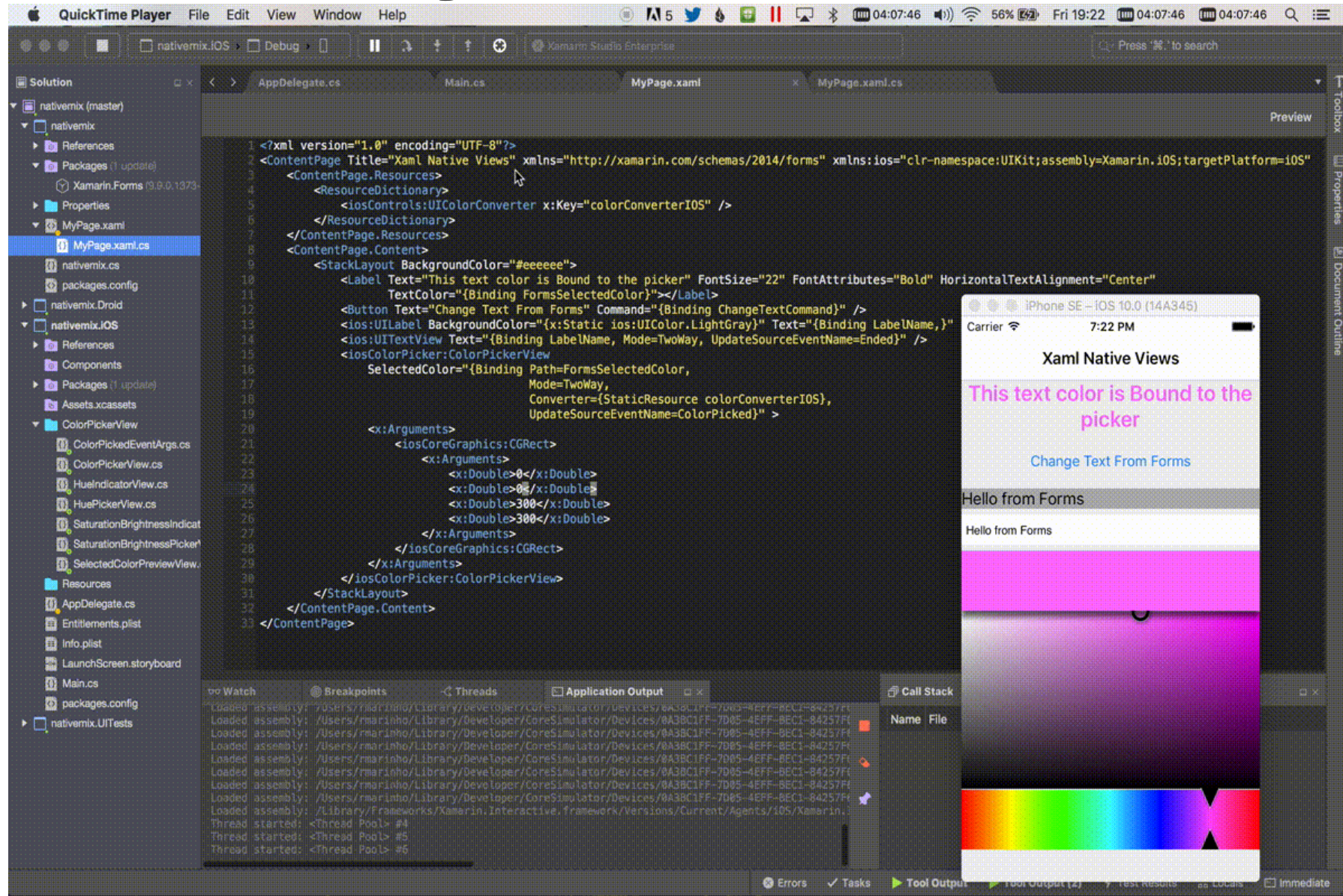
<https://developer.xamarin.com/guides/xamarin-forms/themes/>

# Native Embedding





# Native Embedding



# 10 Minute Break

**Roy Cornelissen**

Lead Consultant, Xpirit

---

[rcornelissen@xpirit.com](mailto:rcornelissen@xpirit.com)

[roycornelissen.wordpress.com](http://roycornelissen.wordpress.com)

[@roycornelissen](https://twitter.com/roycornelissen)