

INTRODUCTION TO XAMARIN MOBILE DEVELOPMENT

Part 2: Building Cross Platform Mobile Applications using Xamarin



xamarin



INTRODUCTION



Chris Key
Senior Consultant

Open Circle Solutions
www.opencirclesolutions.com



SERIES

PART 1: INTRODUCTION TO XAMARIN MOBILE DEVELOPMENT

September 14

PART 2: BUILDING CROSS PLATFORM MOBILE APPLICATIONS USING XAMARIN

Today

PART 3: MOBILE DEV OPS POWERED BY XAMARIN AND MICROSOFT

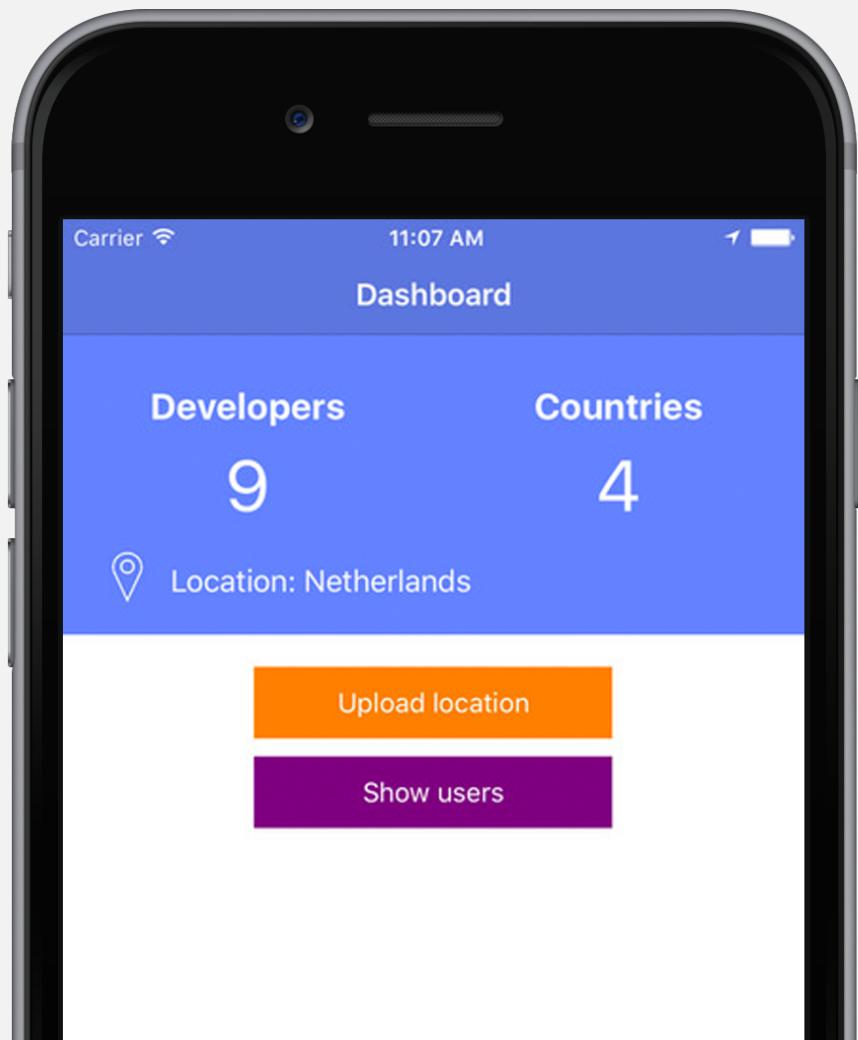
October 12, 2016 – 11:00-12:00 CET

AGENDA

- DEVELOPING CROSS PLATFORM MORE EFFICIENTLY
- XAMARIN FORMS
- DEMO: XAMARIN FORMS
- MVVMCROSS
- DEMO: MVVMCROSS

DEVELOPING CROSS PLATFORM MORE EFFICIENTLY

DEMO APP: XAMLOC



SOURCE FOR DEMO APP

<https://github.com/chriskeynl/xamwebcasts>

DEMO APP: XAMLOC IOS

<http://tinyurl.com/hxyg5fl>

DEMO APP: XAMLOC ANDROID

<http://tinyurl.com/jjcklb5>



COMPONENTS THAT MAKE UP AN APP

XAMLOC

UI

GPS Logic

Localizations

Business rules

Dialogs / notifications

Configuration files

Caching

Push notifications

Navigation

Meta data

Storage

Validators

Services

Models

Background services

Device specific hardware

ADDITIONAL

XAMARIN PLUGINS



UI+APIs

Battery
GPS
Lights
Notifications
Settings
Text To Speech

UI + APIs

Battery
GPS
Lights
Notifications
Settings
Text To Speech

UI + APIs

Battery
GPS
Lights
Notifications
Settings
Text To Speech

XAMARIN PLUGINS

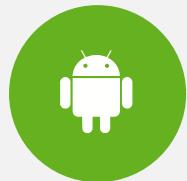
```
Speak("Hello World");
```



AVSpeechSynthesizer

TextToSpeech

SpeechSynthesizer

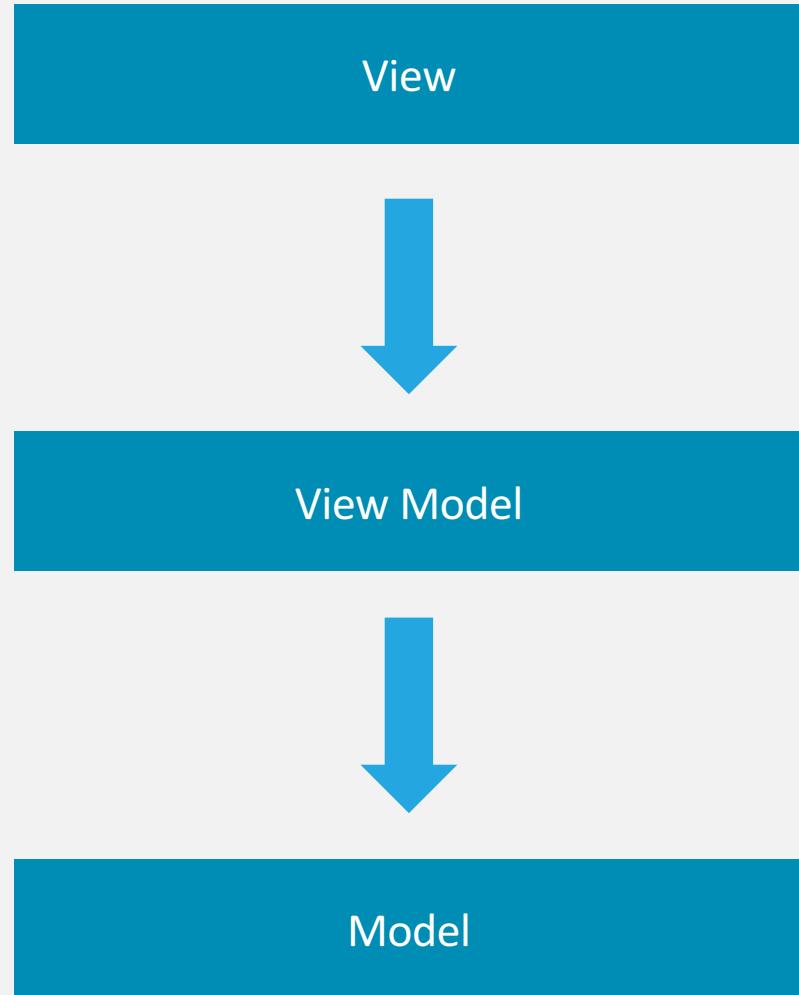


XAMARIN PLUGINS

Common API



MVVM PATTERN



The view is responsible for defining the structure, layout, and appearance of what the user sees on the screen with a limited code-behind that does not contain business logic.

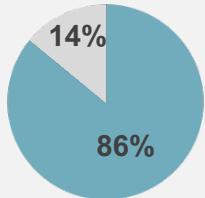
The view model acts as an intermediary between the view and the model, and is responsible for handling the view logic.

The view model acts as an intermediary between the view and the model, and is responsible for handling the view logic.

SHARED CODE STATS

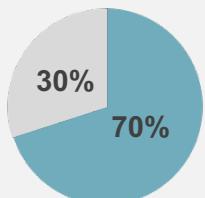
iCircuit

Android

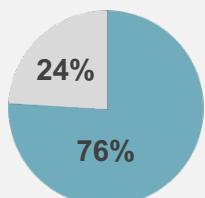


Touch Draw

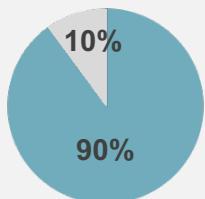
iOS



Mac

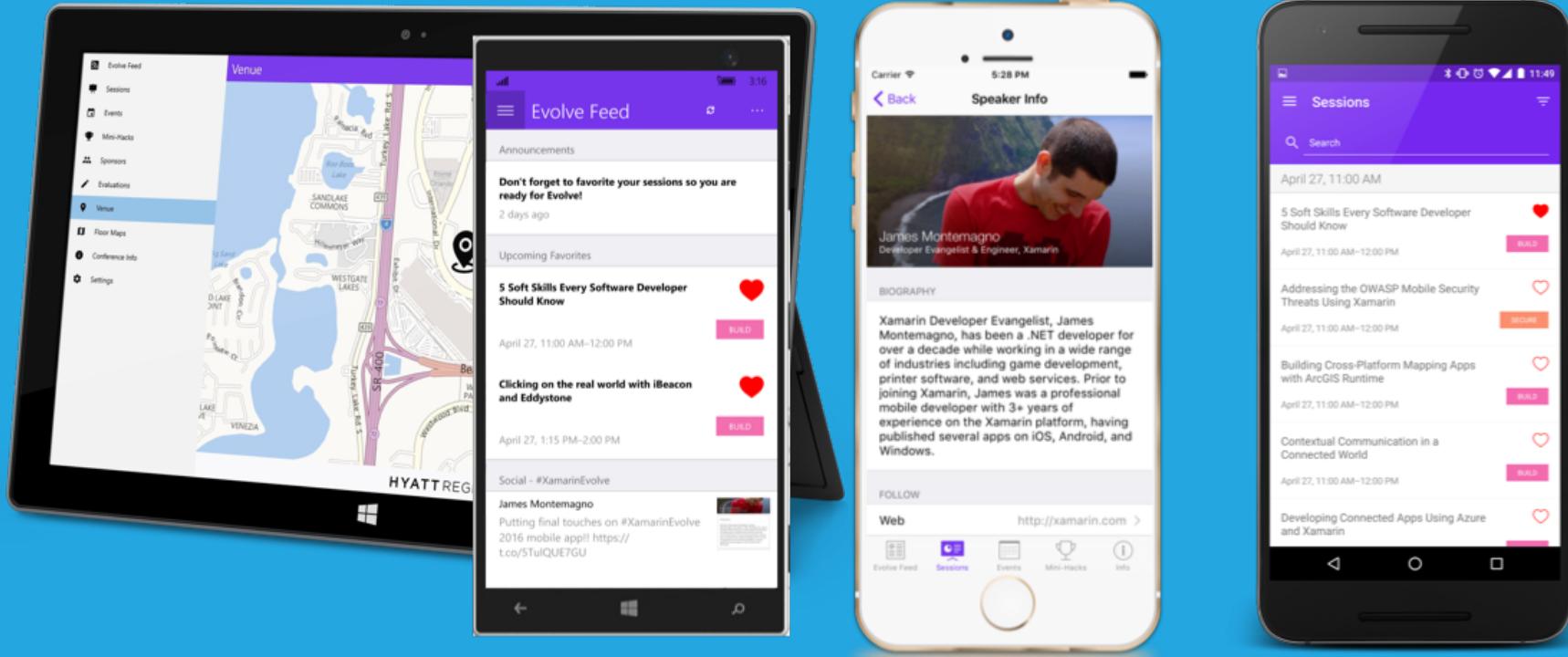


Windows



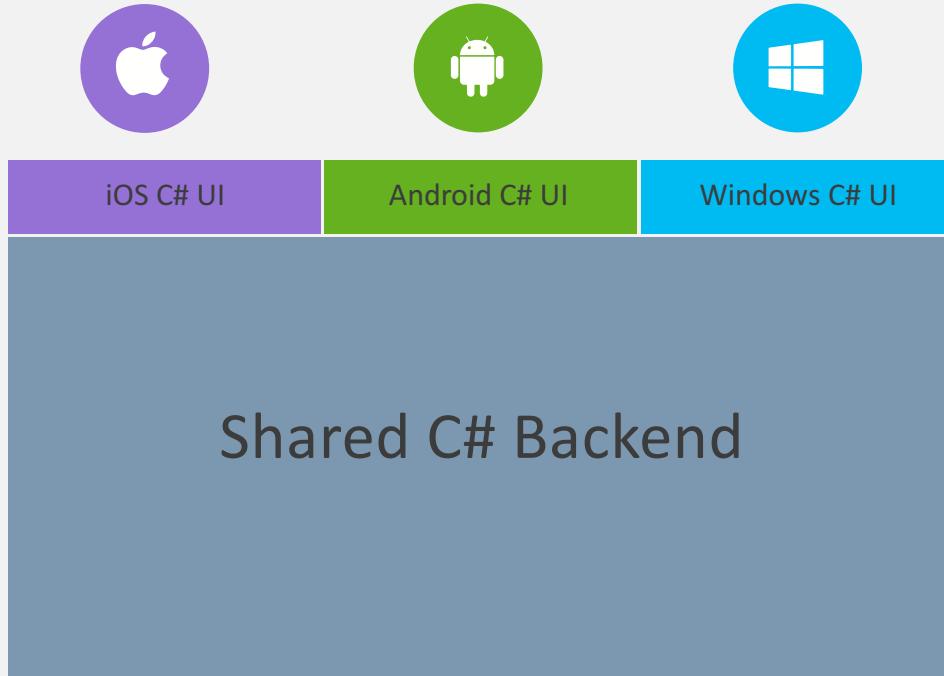
Code Sharing
Stats

XAMARIN FORMS

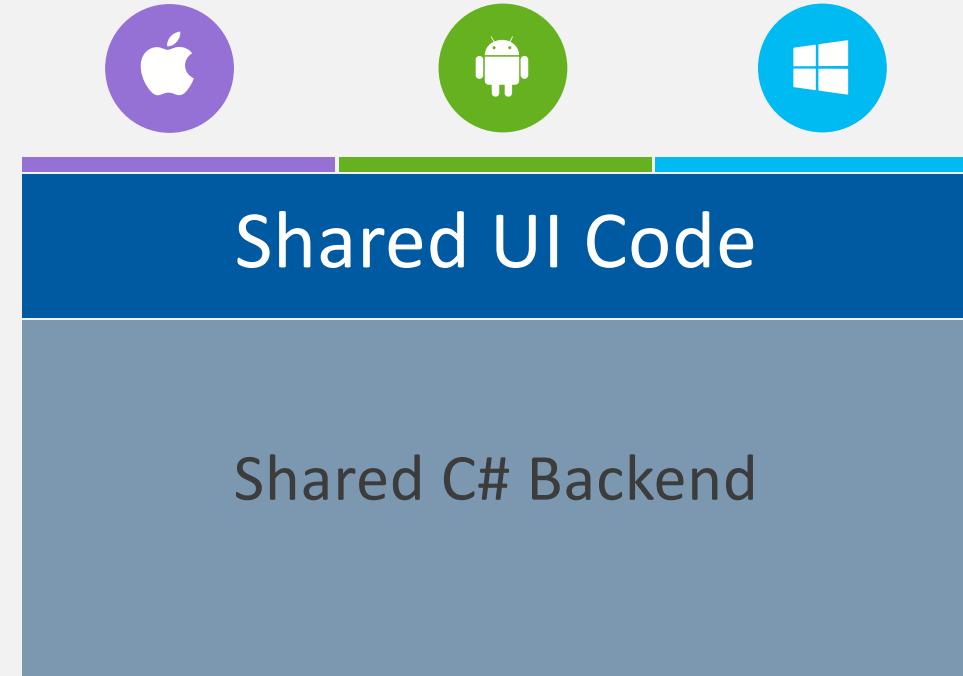


Build native UIs for iOS, Android, and Windows Phone from a single, shared C# codebase.

XAMARIN FORMS

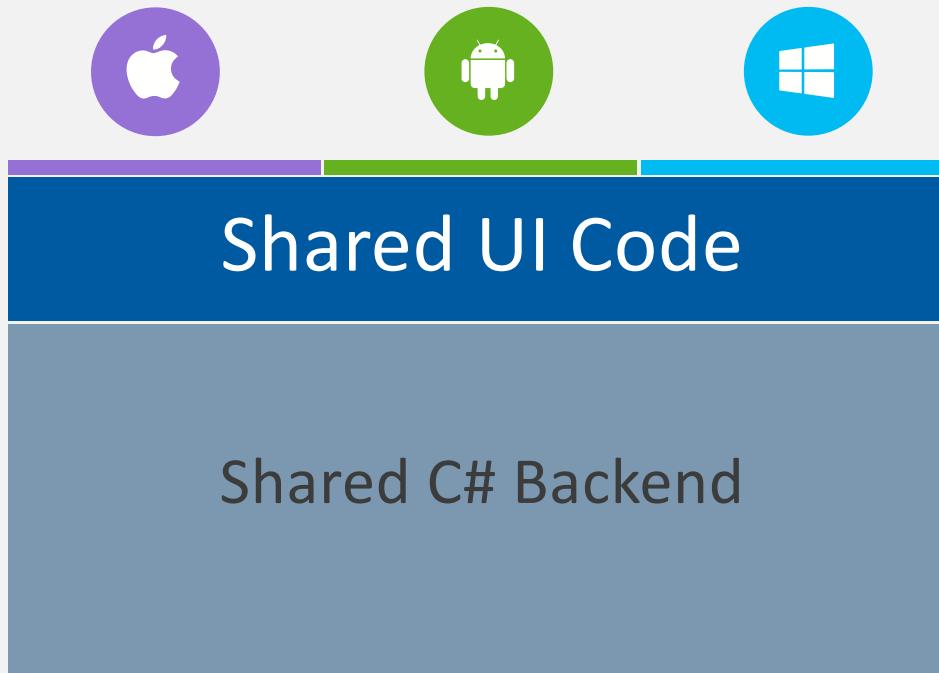


Traditional Xamarin Approach



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

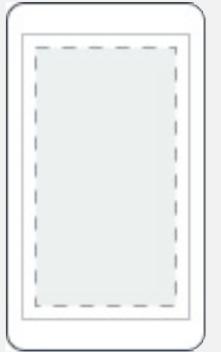
WHAT'S INCLUDED



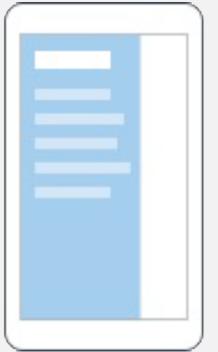
- ✓ 40+ Pages, layouts, and controls
(Build from code behind or XAML)
- ✓ Two-way data binding
- ✓ Navigation
- ✓ Animation API
- ✓ Dependency Service
- ✓ Messaging Center

WHAT'S INCLUDED

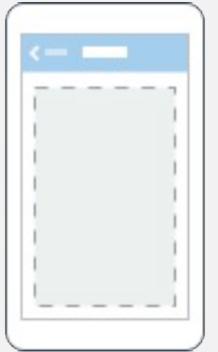
Pages



Content



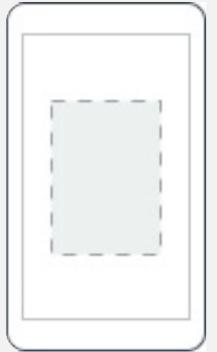
MasterDetail



Navigation

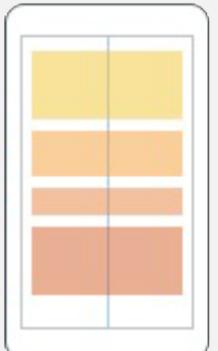


Tabbed

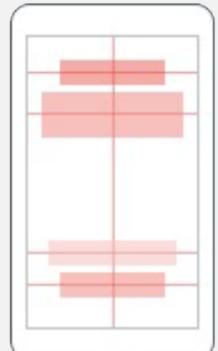


Carousel

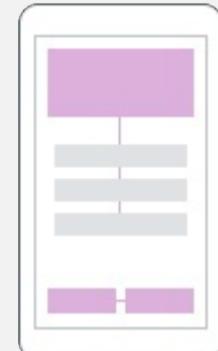
Layouts



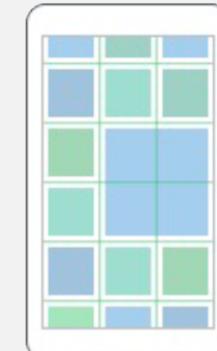
Stack



Absolute



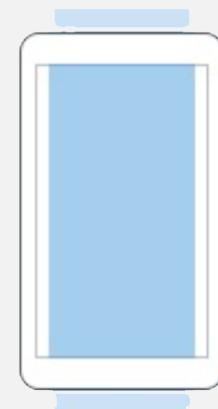
Relative



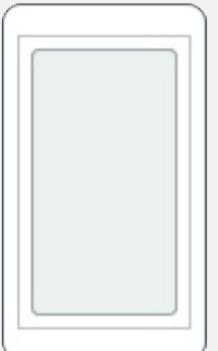
Grid



ContentView



ScrollView



Frame

CONTROLS

ActivityIndicator

BoxView

Button

DatePicker

Editor

Entry

Image

Label

ListView

Map

OpenGLView

Picker

ProgressBar

SearchBar

Slider

Stepper

TableView

TimePicker

WebView

EntryCell

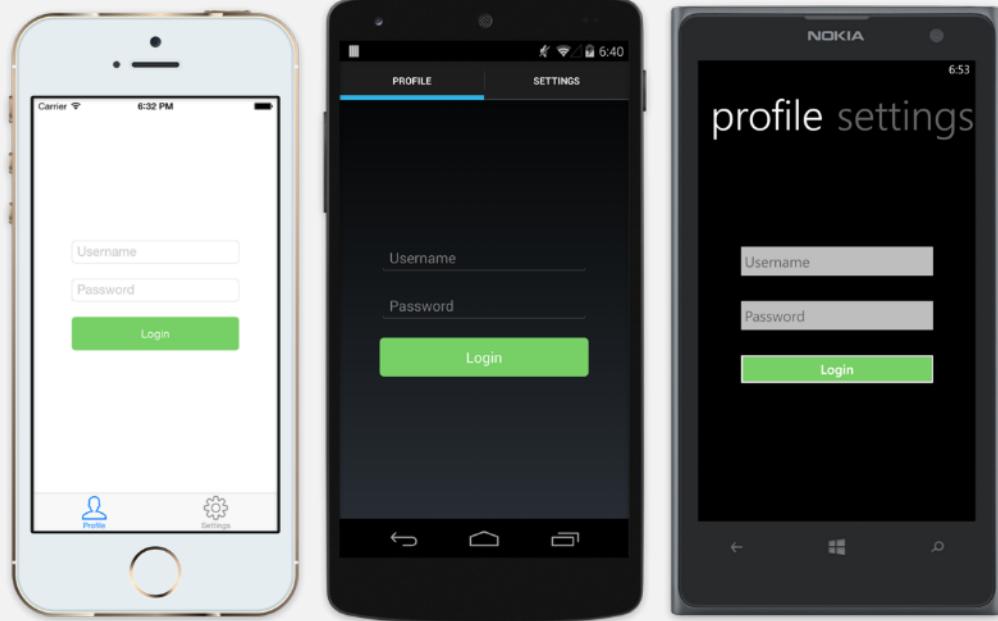
ImageCell

SwitchCell

TextCell

ViewCell

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>
```

NATIVE UI FROM SHARED CODE

```
<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>
```

XAML PREVIEWER

Debug | iPhoneSimulator > iPhone 6s iOS 9.3 Xamarin Studio Business ▲ 7 ⓘ Q Press '%.' to search

Solution App.cs ImageCircleRenderer.cs ImageCircleRenderer.cs CoffeesPage.xaml ⌂ Hide Preview Unit Tests

```
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="White">
19                <Label Text="Made at Home?">
20                    TextColor="White"
21                    VerticalOptions="Center"
22                    HorizontalOptions="Start"
23                    FontSize="16">
24                        <Label.FontSize>
```

Device: Phone Tablet Platform: Android iOS

Made at Home? ADD COFFEE

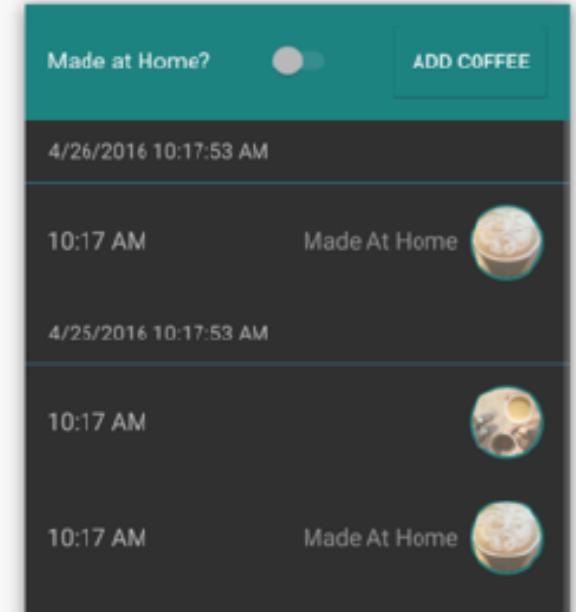
4/26/2016 10:17:53 AM

10:17 AM Made At Home 

4/25/2016 10:17:53 AM

10:17 AM 

10:17 AM Made At Home 



DEMO

XAMARIN FORMS

XAMARIN FORMS RECAP

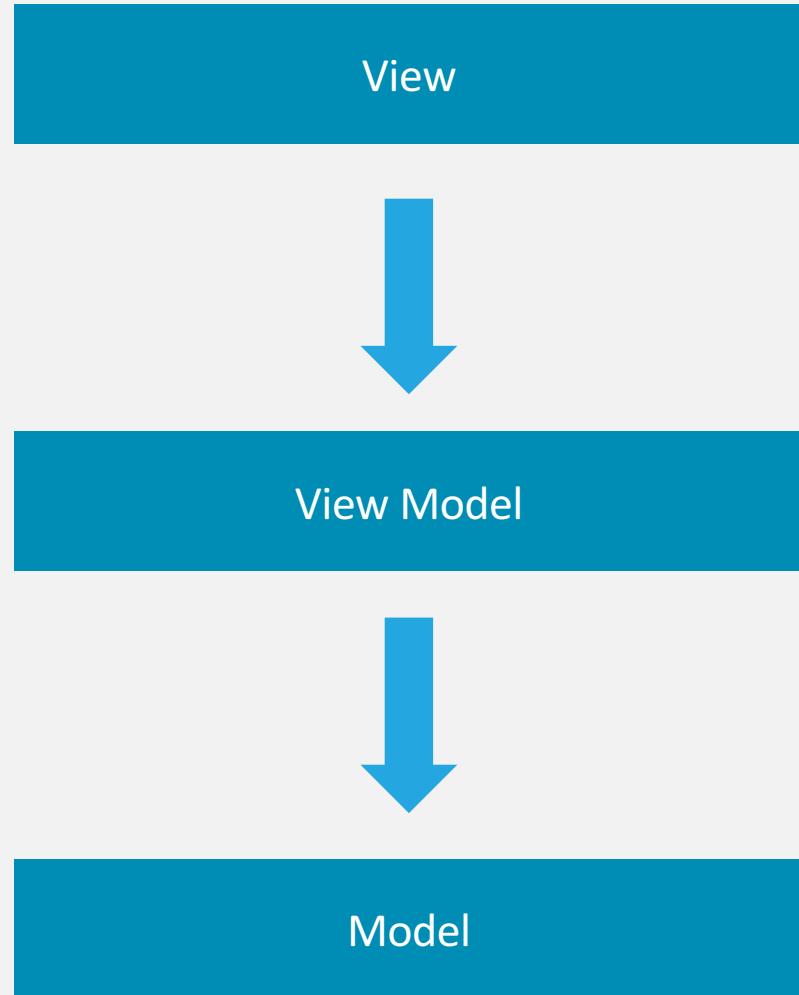
MVVMCROSS



MVVM CROSS

- NATIVE IOS, ANDROID AND WINDOWS UI
- SHARED CODE BASE
- MVVM PATTERN
- PLUGINS
- CONVERTERS

MVVM PATTERN



The view is responsible for defining the structure, layout, and appearance of what the user sees on the screen with a limited code-behind that does not contain business logic.

The view model acts as an intermediary between the view and the model, and is responsible for handling the view logic.

The view model acts as an intermediary between the view and the model, and is responsible for handling the view logic.

MVVMCROSS

CORE

App

Start

View Models

APP

Native application

Setup

Views

Presenter

PLUGINS

Accelerometer

Bookmarks

Color

DownloadCache

Email

Field Binding

File

Json

JsonLocalization

Location

Messenger

MethodBinding

Network

PhoneCall

PictureChooser

ReflectionEx

ResourceLoader

ResxLocalization

SQLite-PCL

Share

SoundEffects

ThreadUtils

Visibility

ViewCell

* Third Party

DEMO

MVVMCROSS

MVVMCROSS RECAP

THAT IS ALL FOR TODAY

ANY QUESTIONS?

EMAIL:

chris.key@opencirclesolutions.nl

LINKED IN:

<https://www.linkedin.com/in/keychris>

RESOURCES

SLIDES AND SOURCE FOR DEMO APP

<https://github.com/chriskeynl/xamwebcasts>

XAMARIN RESOURCES

https://developer.xamarin.com/guides/cross-platform/getting_started

DEMO APP: XAMLOC IOS

<http://tinyurl.com/hxyg5f1>

DEMO APP: XAMLOC ANDROID

<http://tinyurl.com/jjcklb5>