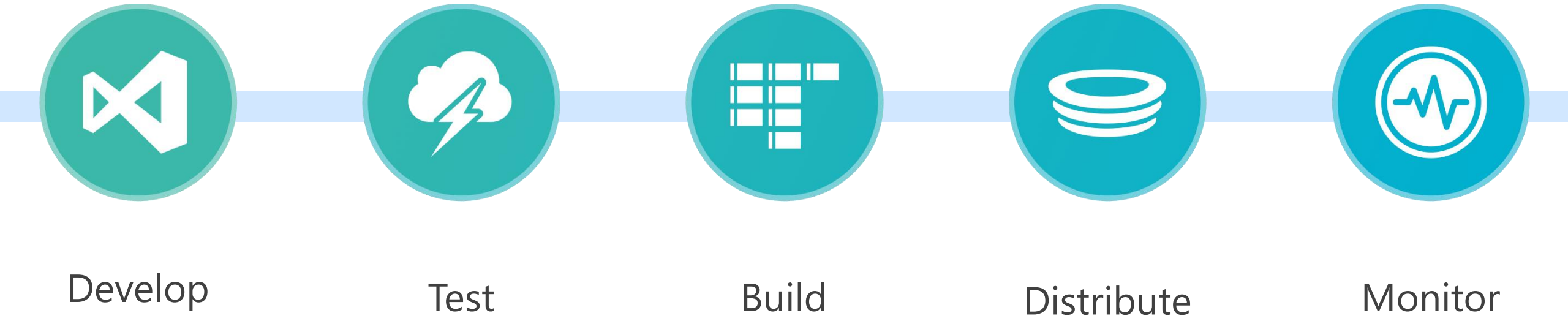
Native iOS, Android, Mac, & Windows Mobile Apps in C# with Xamarin

Joshua Joshua
@joshuajsmart

Xamarin

- Introduction to Xamarin
- Cross Platform UI with Xamarin.Forms

Xamarin – Your Complete Mobile Solution



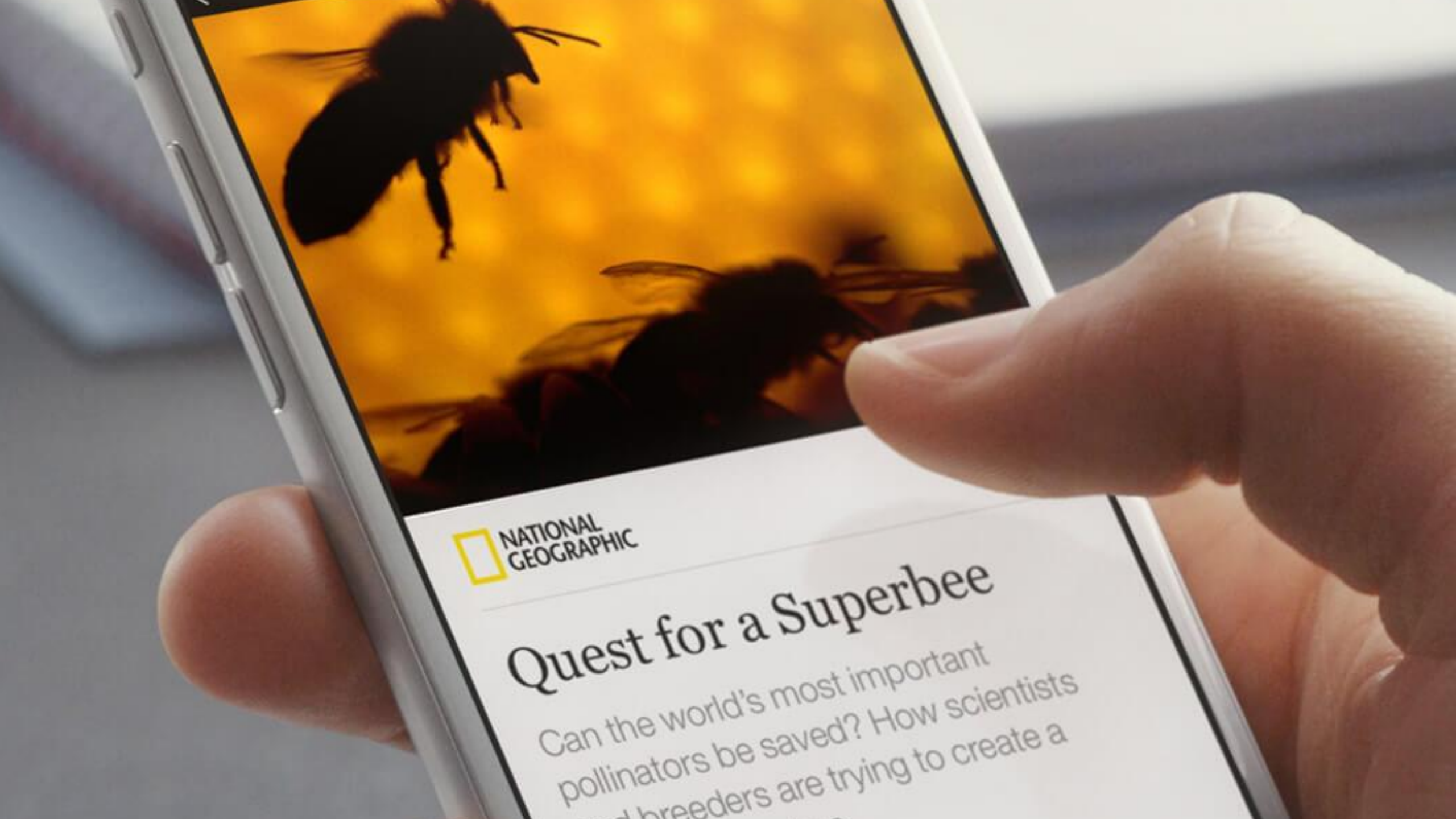
Develop

What is Native?

?

?

?


NATIONAL GEOGRAPHIC
Quest for a Superbee
Can the world's most important pollinators be saved? How scientists
breeders are trying to create a

"Not only is native mobile code winning for app development, but with things like Instant Articles, native is making the browser-based web look like a relic even just for publishing *articles*."

-John Gruber

http://daringfireball.net/2015/05/facebook_instant_articles

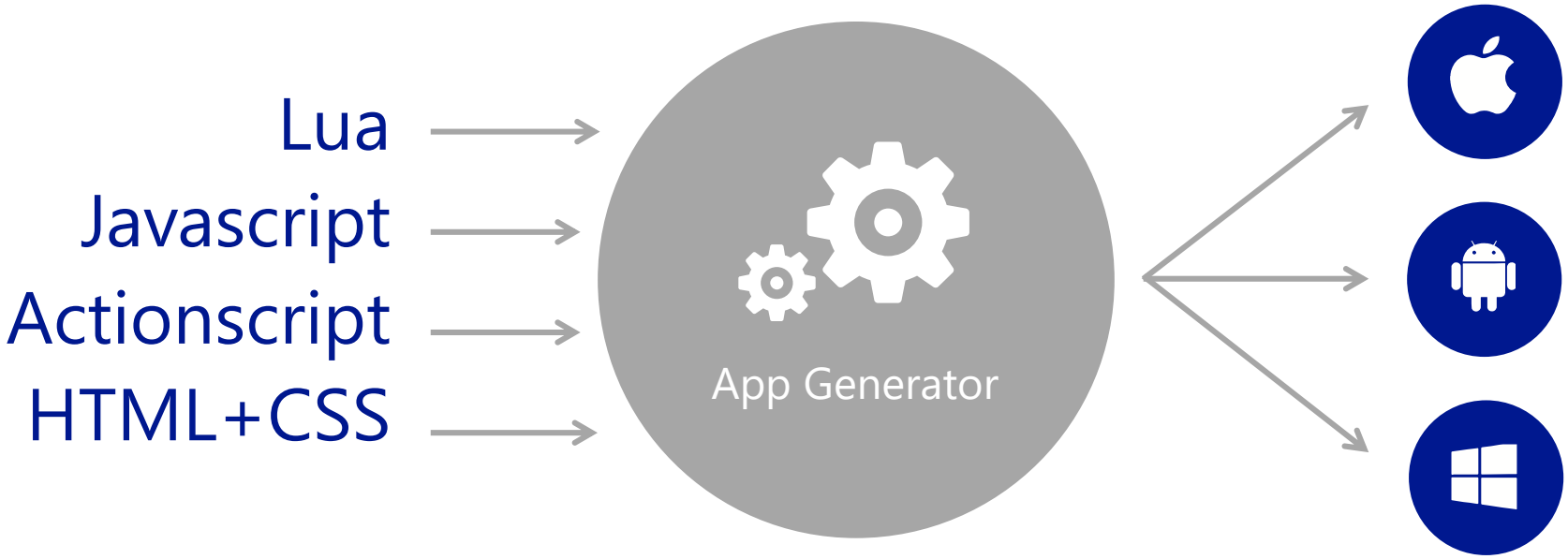
Architecting Mobile Apps

Silo Approach

iOS	Android	Windows
Objective-C Xcode	Java Android Studio	C# Visual Studio

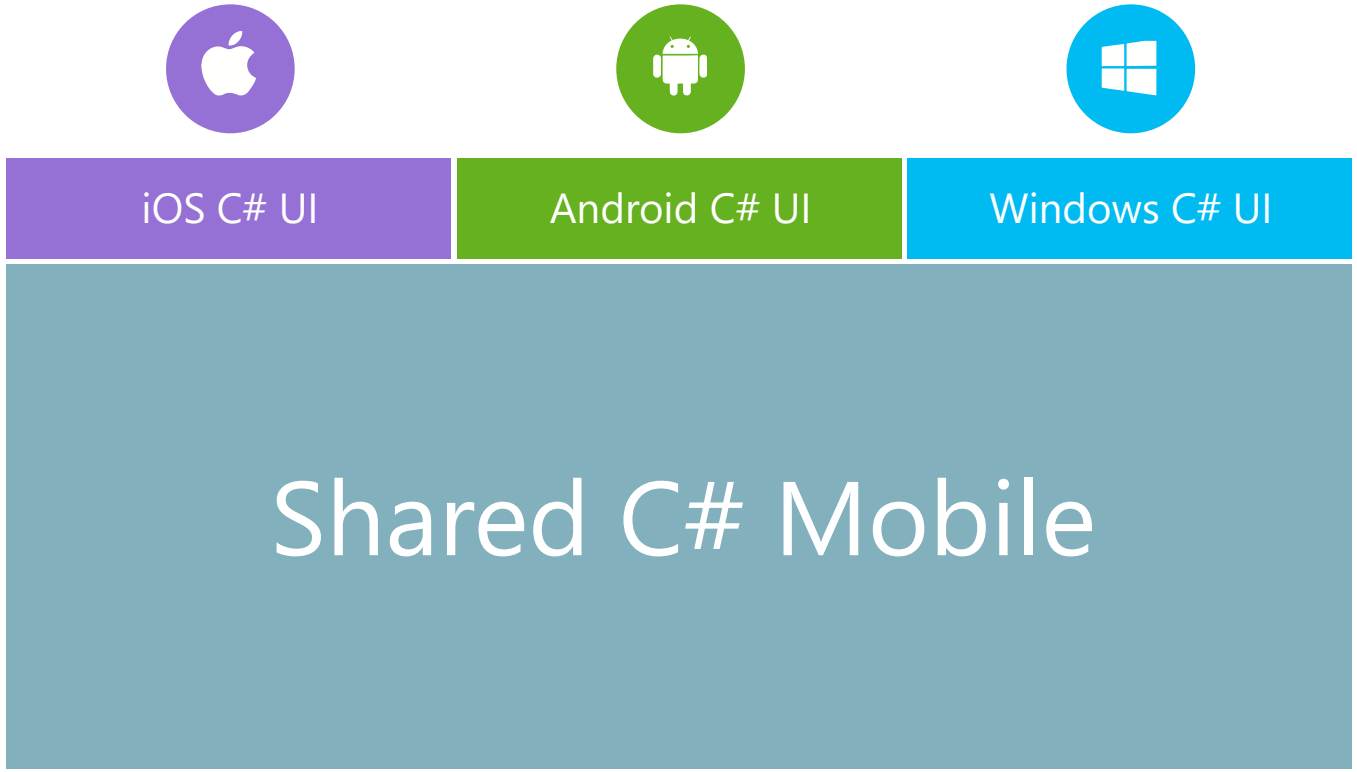
No shared code • Many languages & development environments • Multiple teams

Write Once, Run Anywhere



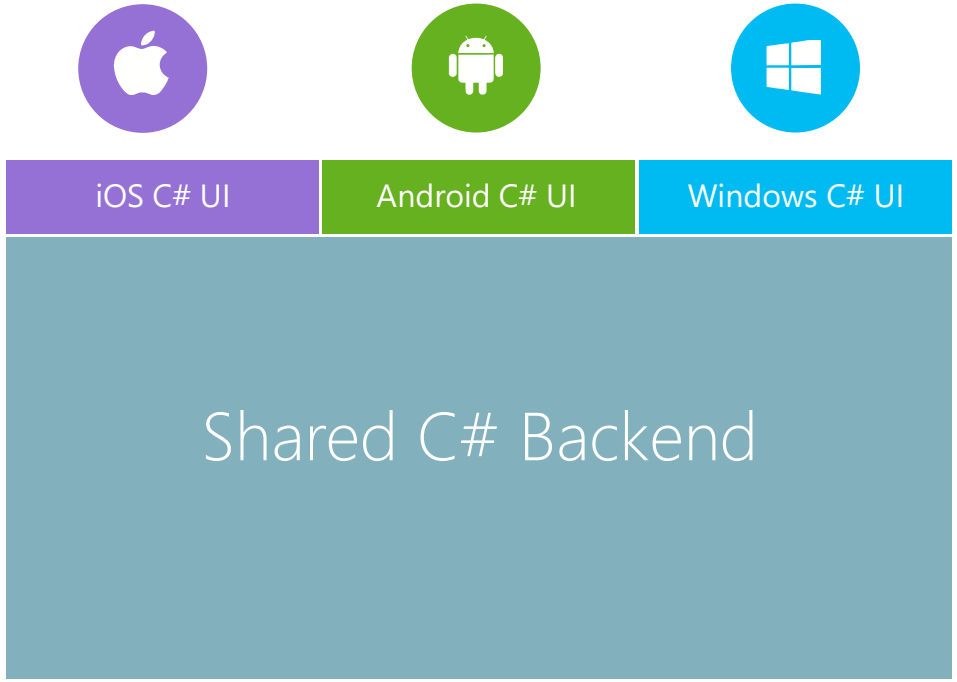
Limited native API access • Slow performance • Poor user experience

Xamarin's Unique Approach

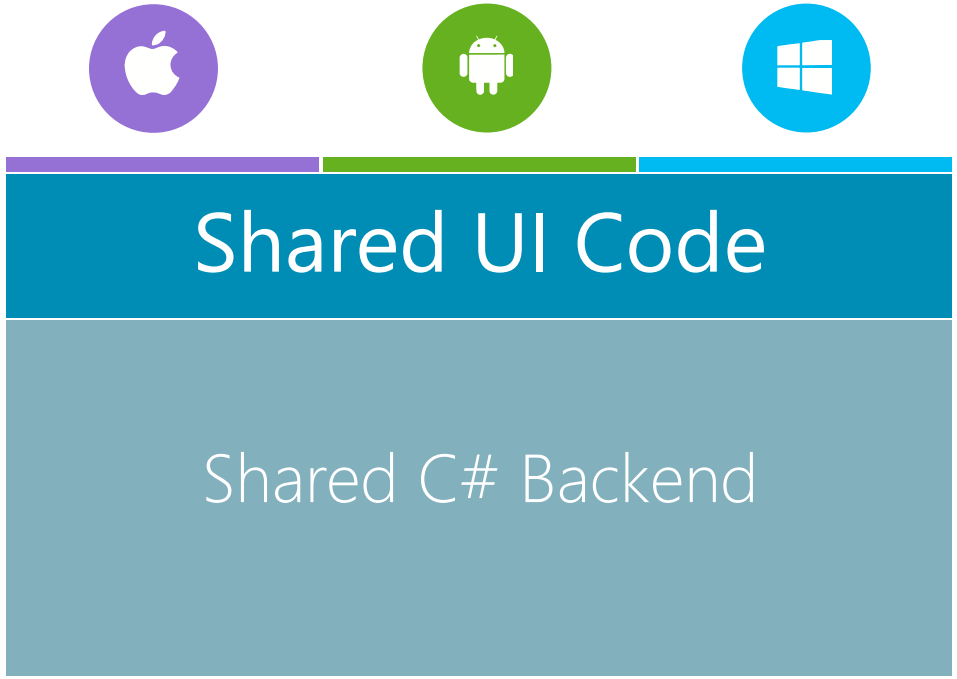


Shared C# codebase • 100% native API access • High performance

Xamarin + Xamarin.Forms

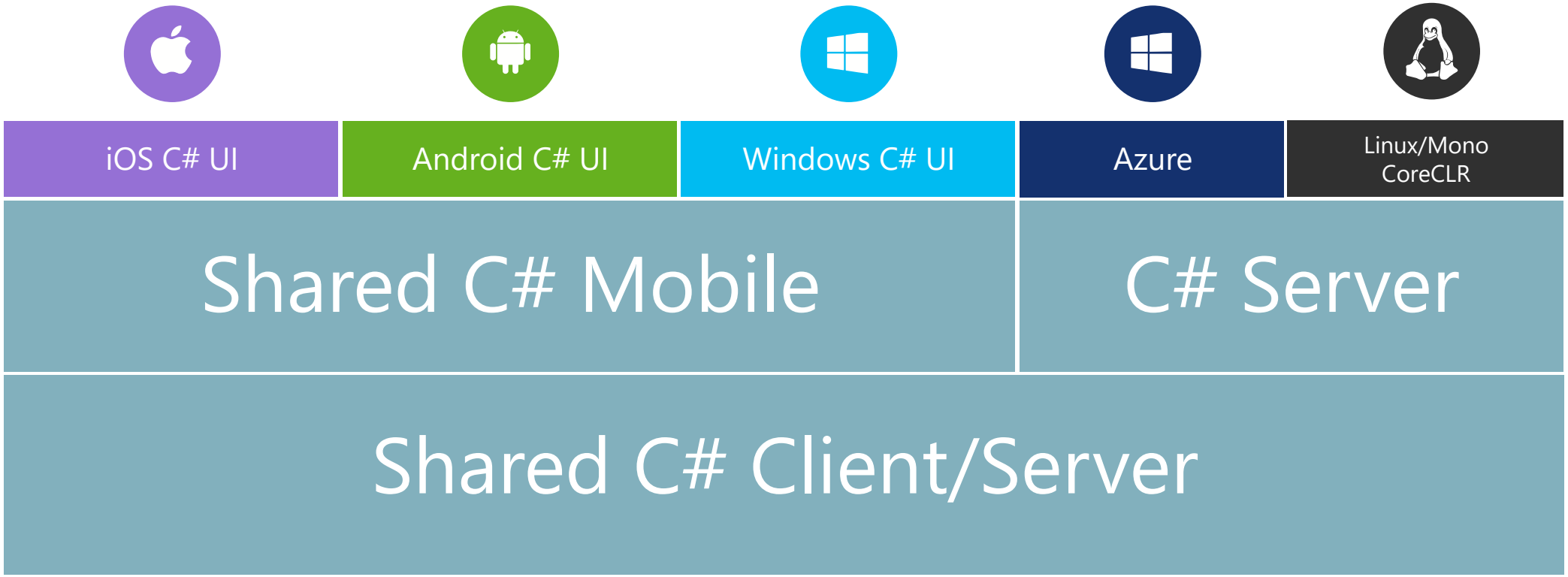


Traditional Xamarin Approach



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

Xamarin's Unique Approach



Shared C# codebase • 100% native API access • High performance

Why Xamarin

Xamarin is included in Visual Studio

.NET

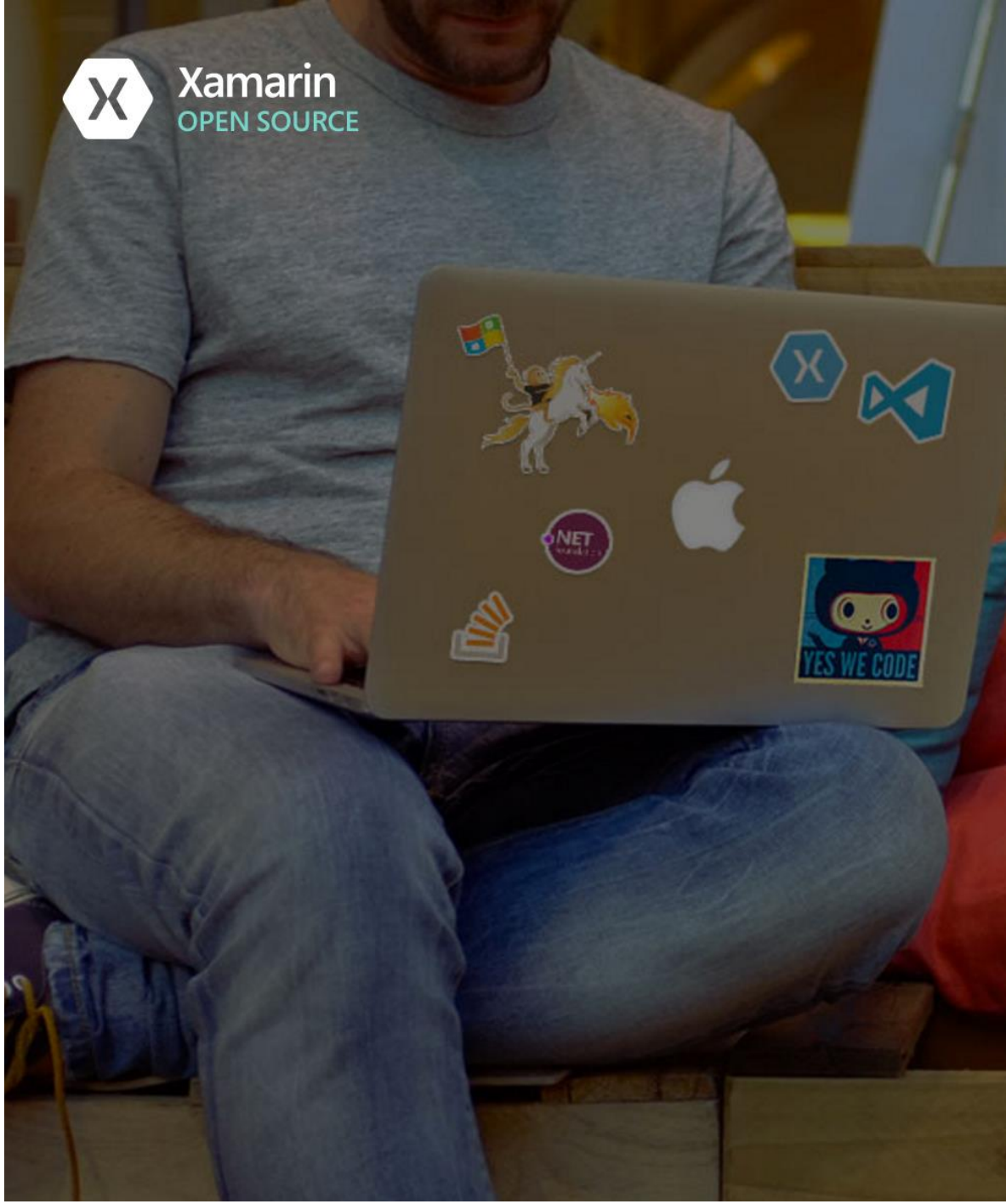
NATIVE EVERYWHERE OPEN SOURCE

Open Source – open.xamarin.com

Open Source

- Xamarin.iOS
- Xamarin.Mac
- Xamarin.Android
- Xamarin.Forms
- Bindings & Plugins

- Getting Started
- Contribution Guides



Write Everything in C#

C# Now Runs on 2.6+ Billion Devices

Why C#?

C# is Awesome

```
from p in Table<Person> ()
    where p.ID == id
    select p;
```

LINQ Support

```
var doc = XDocument.Load(url);
foreach(var item in doc.Root.Elements()) {
    var text = item.Value;
}
```

Work with XML Easily

```
button.TouchUpInside += (s, o) => {
    message.Text = "Hello!";
};
```

Event Handling & Delegates

JSON Made Easy

Json.NET offers simple conversations to/from JSON strings and .NET objects:

```
public class Person
{
    public string Name { get; set; }
    public DateTime Birthday { get; set; }
}
var person = new Person { Name = "Bob", Birthday = new DateTime (1987, 2, 2) };
var output = Newtonsoft.Json.JsonConvert.SerializeObject (person);

person = Newtonsoft.Json.JsonConvert.DeserializeObject<Person> (output);
Console.WriteLine ("{0} - {1}", person.Name, person.Birthday);
```

See the Different – Classes & Methods

```objc
// Objective-C
@interface Person : NSObject
@property (strong, nonatomic) NSString *name;
@end

@implementation Person
- (id)initWithName:(NSString *)name {
    self = [super init];
    if (self) {
        self.name = name;
    }
    return self;
}

+ (NSArray *)getNames {
    NSArray *people = @[
      [[Person alloc] initWithName:@"David"],
      [[Person alloc] initWithName:@"Vinicius"],
      [[Person alloc] initWithName:@"Serena"],
    ];
    NSMutableArray *names = [NSMutableArray array];
    [people enumerateObjectsUsingBlock:^(Person *person,
                                         NSUInteger idx,
                                         BOOL *stop) {
        [names addObject:person.name];
    }];
    return names;
}
@end
```

```csharp
// C# with Xamarin
class Person : NSObject {
    public string Name { get; set; }

    public static string[] GetNames() {
        var people = new[] {
            new Person { Name="David" },
            new Person { Name="Vinicius" },
            new Person { Name="Serena" },
        };
        return people.Select(person => person.Name).ToArray();
    }
}
```

Async/Await

```
public async Task ExecuteGetPodcastsCommand()
{
  if (IsBusy)
    return;

  try
  {
    IsBusy = true;
    var client = new HttpClient();
    // Request from server podcast xml
    var podcastString = await client.GetStringAsync(PodcastUrl);

    // Parse Xml into data model and load into list
    var casts = await ParseXml(podcastString);

    foreach (var cast in casts)
    {
      Podcasts.Add(cast);
      FilteredPodcasts.Add(cast);
    }

  }
```

Write Beautiful & Maintainable Code

Simplifying Native Methods with Async/Await

Objective-C

```
[UIView animateWithDuration:0.2
     animations:^{view.alpha = 0.0;}
     completion:^(BOOL finished){ [view removeFromSuperview]; }];
```

C# with Xamarin

```
//Animate alpha to 0 asynchronously.
//await animation and then remove from superview
bool success = await UIView.AnimateAsync(2, () => { UIView.Alpha = 0;});
view.RemoveFromSuperview ();
```

See the Difference – Android ItemClick

Java

```java
listView.setOnItemClickListener(new OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {

    // Value of item
    String  itemValue    = (String) listView.getItemAtPosition(position);

    // Show Toats
    Toast.makeText(getActivity(),"Position :"+ position + "  ListItem : "
                                + itemValue , Toast.LENGTH_LONG).show();
    }

});
```

C# with Xamarin

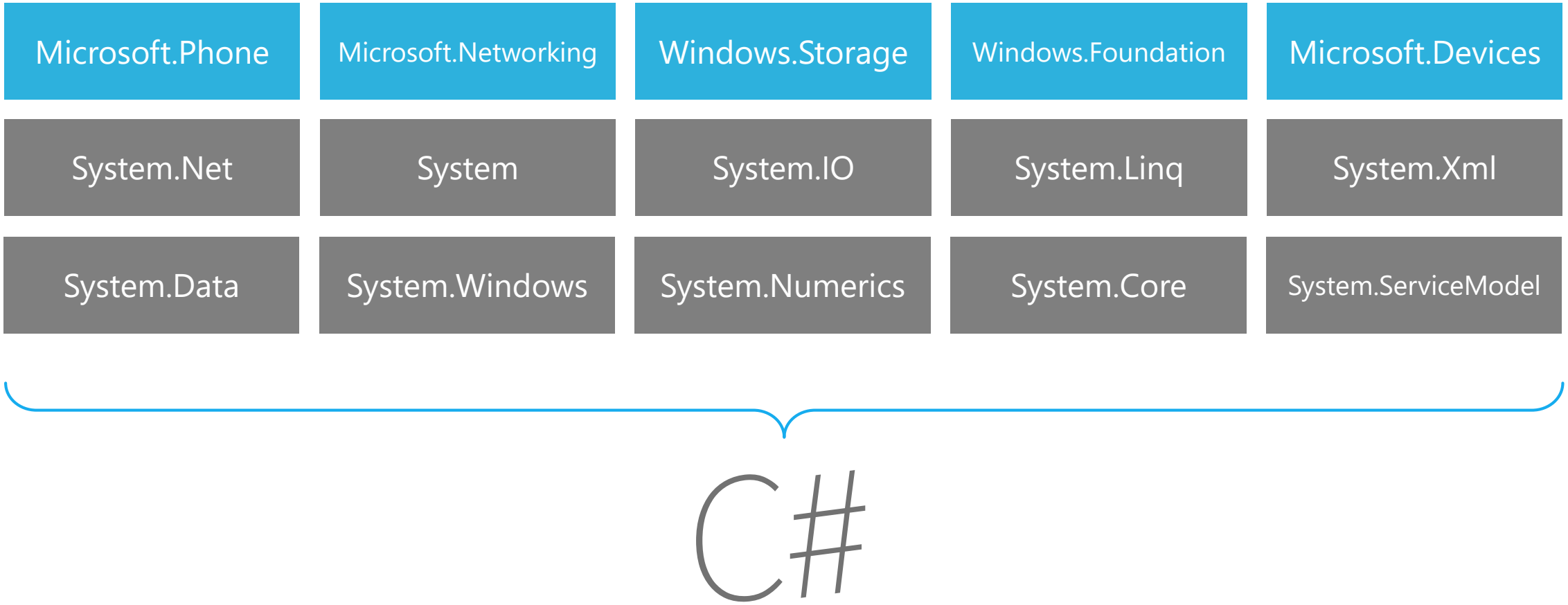
```csharp
listView.ItemClick += (sender, args) => {

    // Value of item
    var itemValue = (string)listView.GetItemAtPosition(args.Position);

    //Show Toast
    Toast.MakeText(this, string.Format("Postition: {0} ListItem: {1}",
                                    args.Position, itemValue), ToastLength.Long).Show();

};
```

C# & Async with Xamarin

```csharp
listView.ItemClick += async (sender, args) => {

    // Value of item
    var itemValue = (string)listView.GetItemAtPosition(args.Position);

    //Show Toast
    Toast.MakeText(this, string.Format("Postition: {0} ListItem: {1}",
                                    args.Position, itemValue), ToastLength.Long).Show();

};
```

How Xamarin Works

Windows APIs



iOS – 100% API Coverage

MapKit	UIKit	iBeacon	CoreGraphics	CoreMotion
System.Net	System	System.IO	System.Linq	System.Xml
System.Data	System.Windows	System.Numerics	System.Core	System.ServiceModel

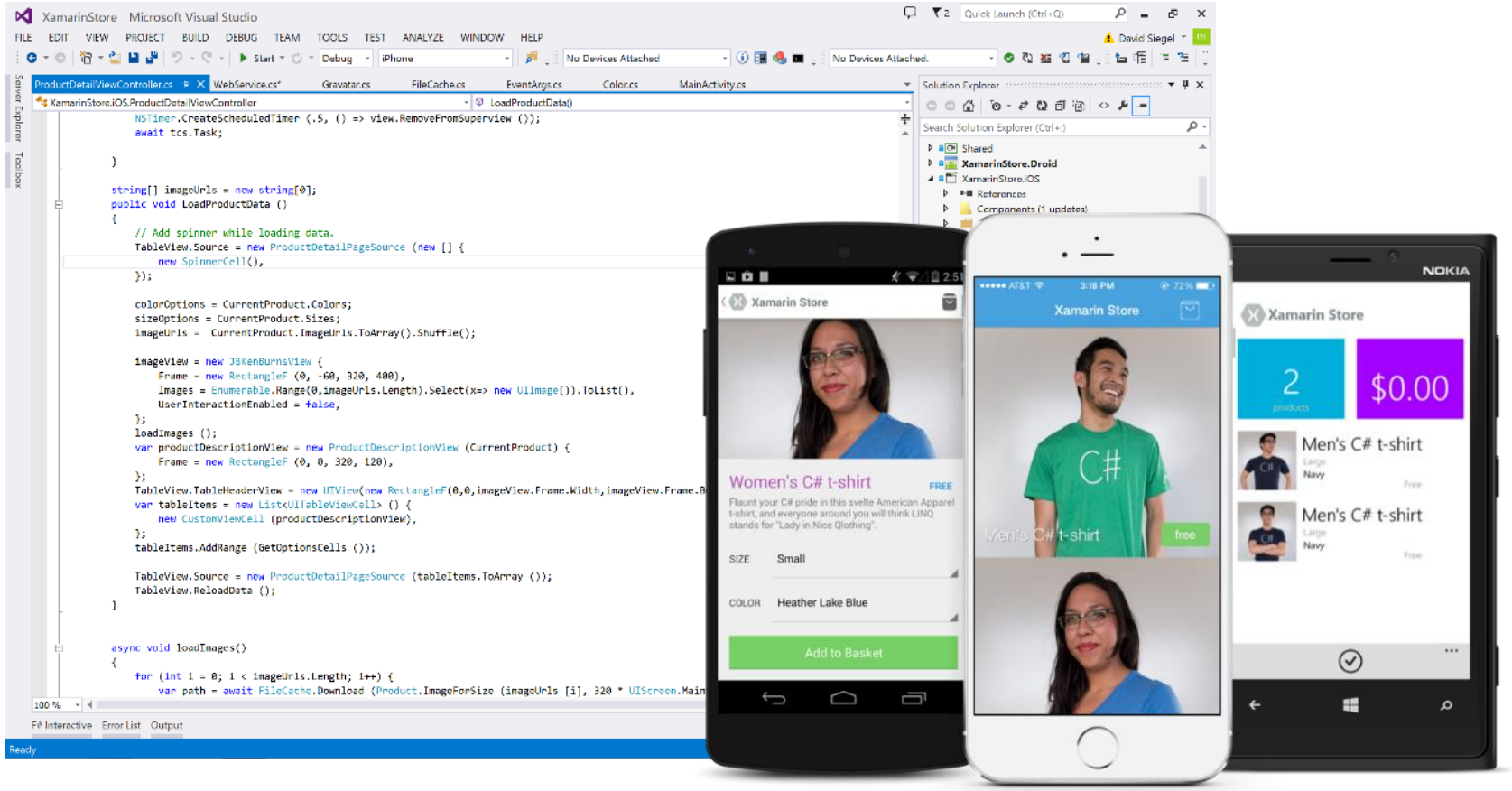
C#

Android – 100% API Coverage

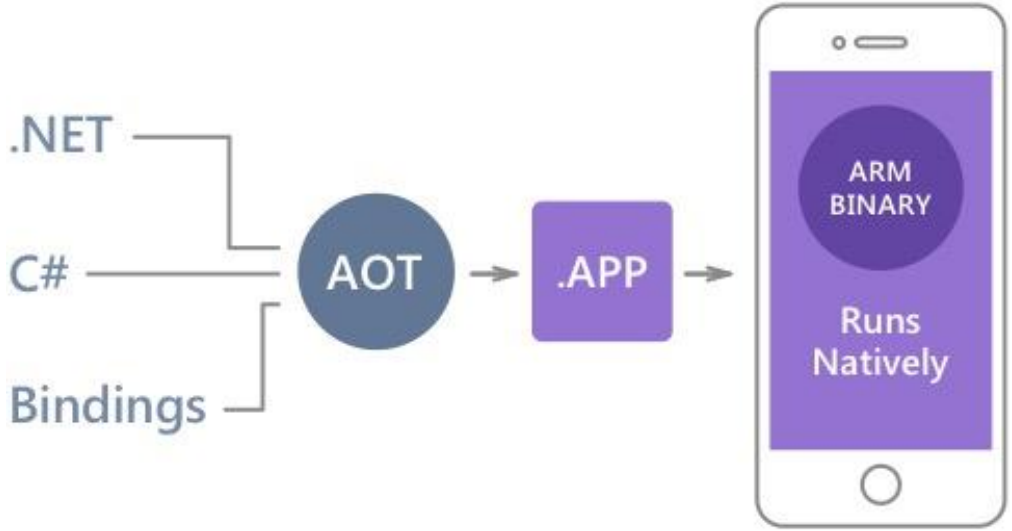
Text-to-speech	ActionBar	Printing Framework	Renderscript	NFC
System.Net	System	System.IO	System.Linq	System.Xml
System.Data	System.Windows	System.Numerics	System.Core	System.ServiceModel

C#

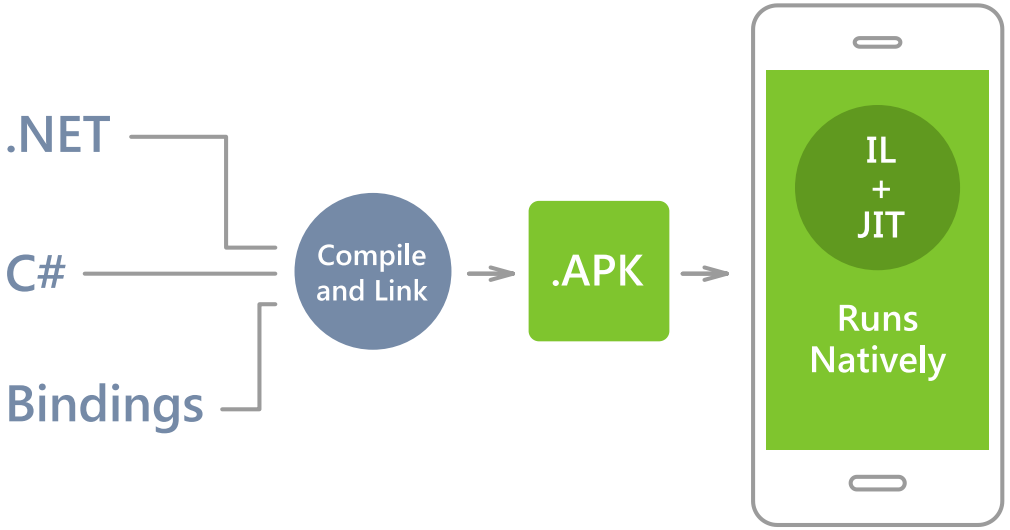
Anything you can do in Objective-C, Swift, or Java can be done **in C# and Visual Studio with Xamarin**.



Native Performance



Xamarin.iOS does full Ahead Of Time (AOT) compilation to produce an ARM binary for Apple's App Store.



Xamarin.Android takes advantage of Just In Time (JIT) compilation on the Android device.

✓Always Up-to-Date

Same-day support:

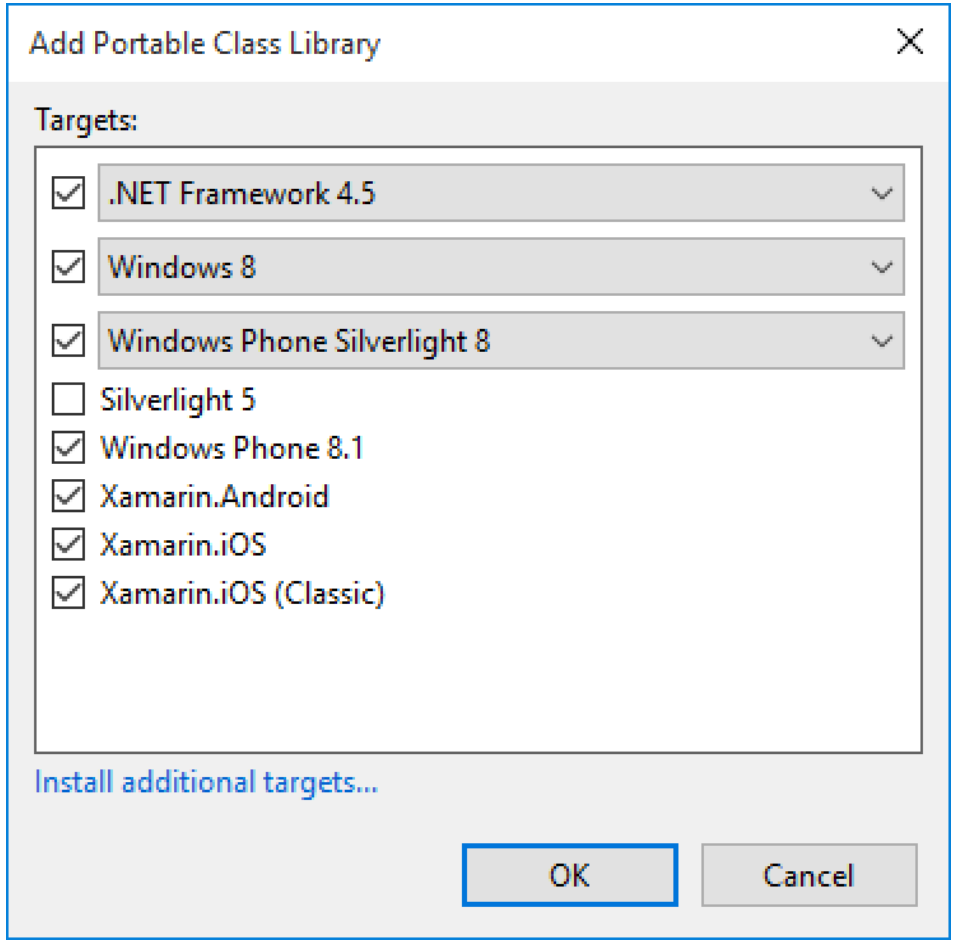
- iOS 5
- iOS 6
- iOS 7
- iOS 7.1
- iOS 8
- iOS 9

Full support for:

- Apple Watch
- Apple TV
- Google Glass
- Android Wear
- Amazon Fire TV
- and much more

Sharing Code

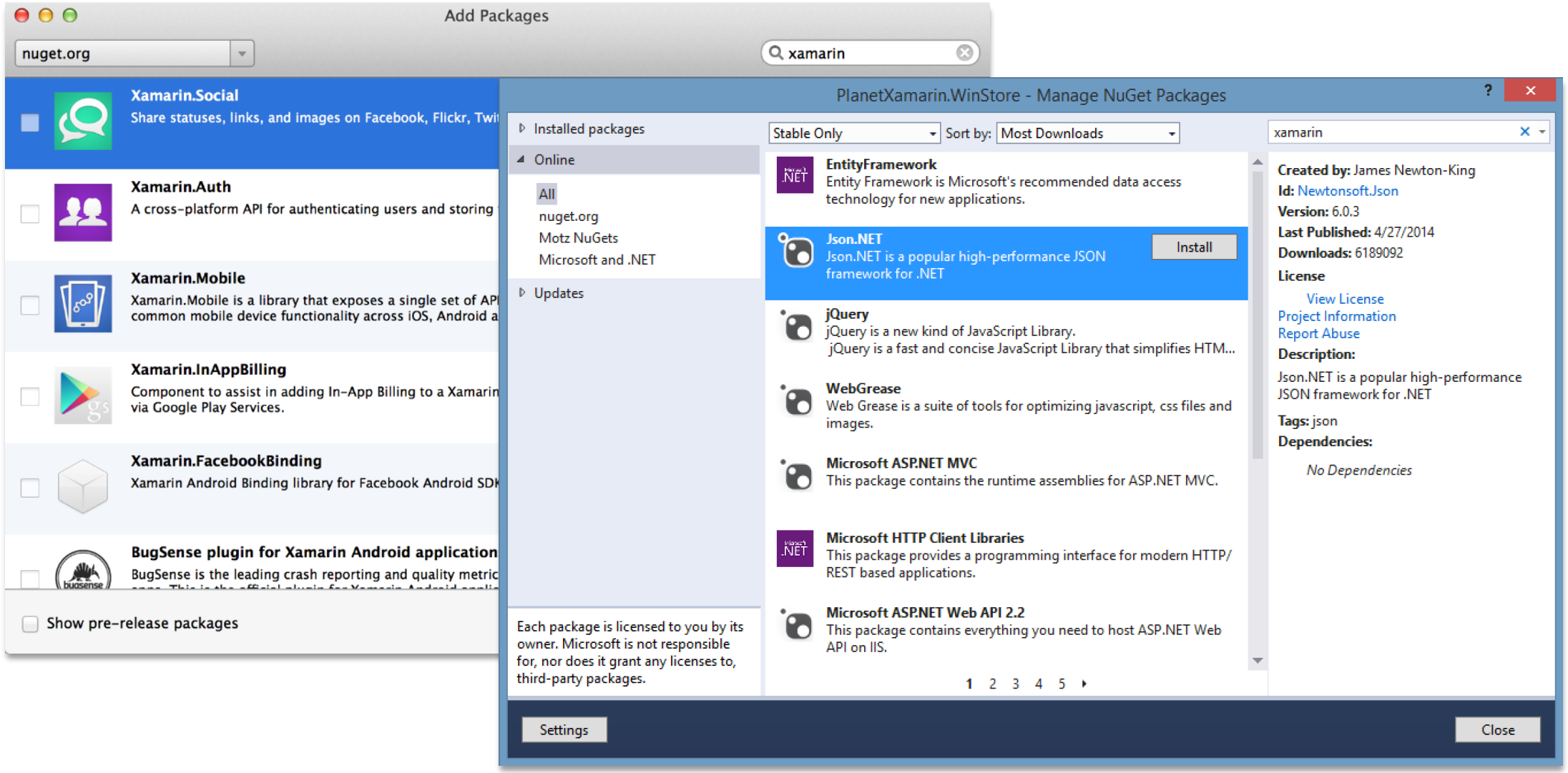
Portable Class Libraries



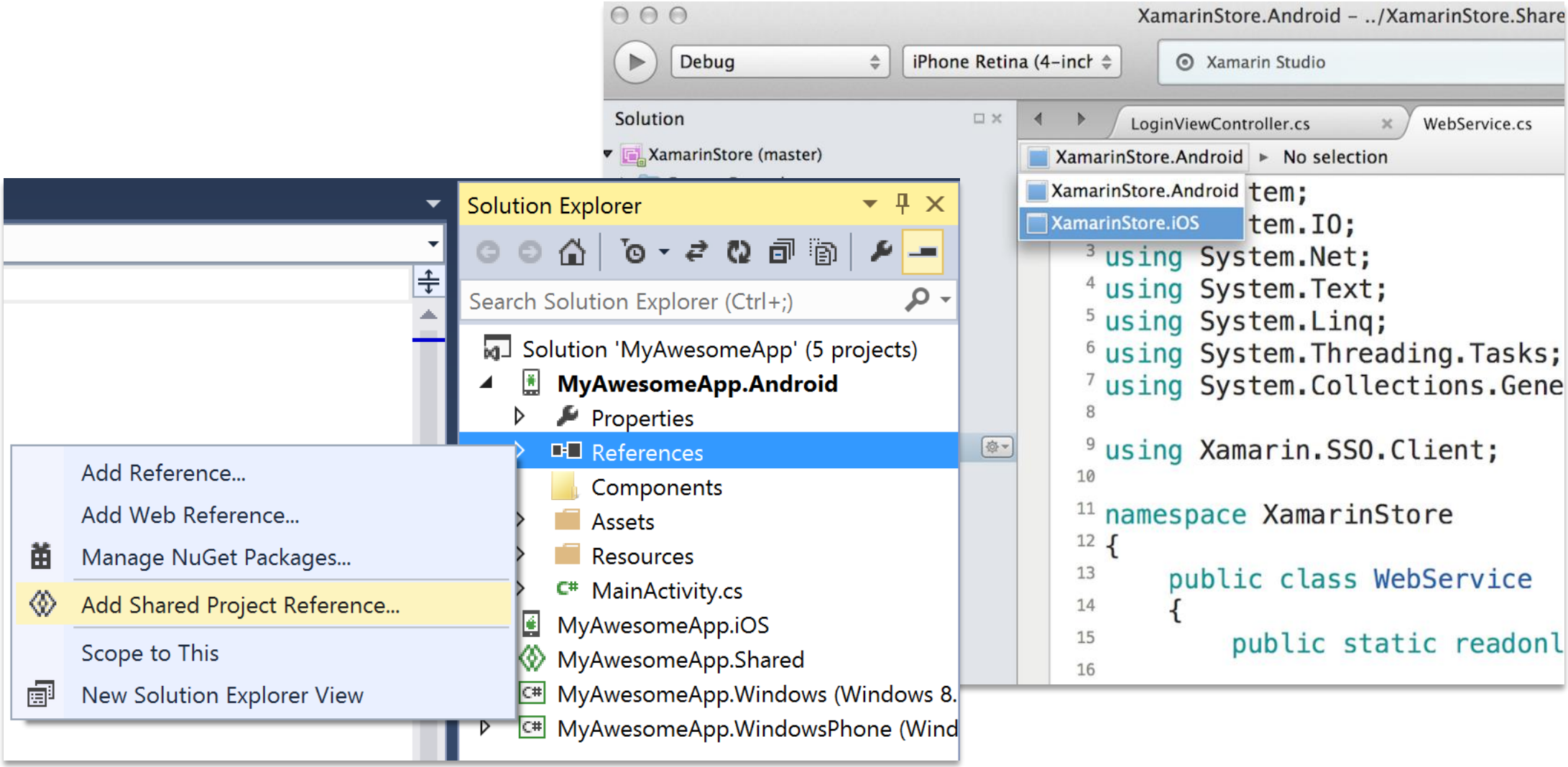
1 Assembly
Multiple Platforms

Including:
Xamarin.iOS and Xamarin.Android

NuGet


Add Packages
nuget.org
xamarin
Xamarin.Social
Share statuses, links, and images on Facebook, Flickr, Twi
Xamarin.Auth
A cross-platform API for authenticating users and storing
Xamarin.Mobile
Xamarin.Mobile is a library that exposes a single set of AP
common mobile device functionality across iOS, Android a
Xamarin.InAppBilling
Component to assist in adding In-App Billing to a Xamarin
via Google Play Services.
Xamarin.FacebookBinding
Xamarin Android Binding library for Facebook Android SD
BugSense plugin for Xamarin Android application
BugSense is the leading crash reporting and quality metric
Show pre-release packages
PlanetXamarin.WinStore - Manage NuGet Packages
Installed packages
Online
All
nuget.org
Motz NuGets
Microsoft and .NET
Updates
Stable Only
Sort by: Most Downloads
xamarin
EntityFramework
Entity Framework is Microsoft's recommended data access technology for new applications.
Json.NET
Install
Json.NET is a popular high-performance JSON framework for .NET
jQuery
jQuery is a new kind of JavaScript Library.
jQuery is a fast and concise JavaScript Library that simplifies HTM...
WebGrease
Web Grease is a suite of tools for optimizing javascript, css files and images.
Microsoft ASP.NET MVC
This package contains the runtime assemblies for ASP.NET MVC.
Microsoft HTTP Client Libraries
This package provides a programming interface for modern HTTP/REST based applications.
Microsoft ASP.NET Web API 2.2
This package contains everything you need to host ASP.NET Web API on IIS.
1 2 3 4 5
Each package is licensed to you by its owner. Microsoft is not responsible for, nor does it grant any licenses to, third-party packages.
Created by: James Newton-King
Id: Newtonsoft.Json
Version: 6.0.3
Last Published: 4/27/2014
Downloads: 6189092
License
View License
Project Information
Report Abuse
Description:
Json.NET is a popular high-performance JSON framework for .NET
Tags: json
Dependencies:
No Dependencies
Settings
Close

Shared Projects

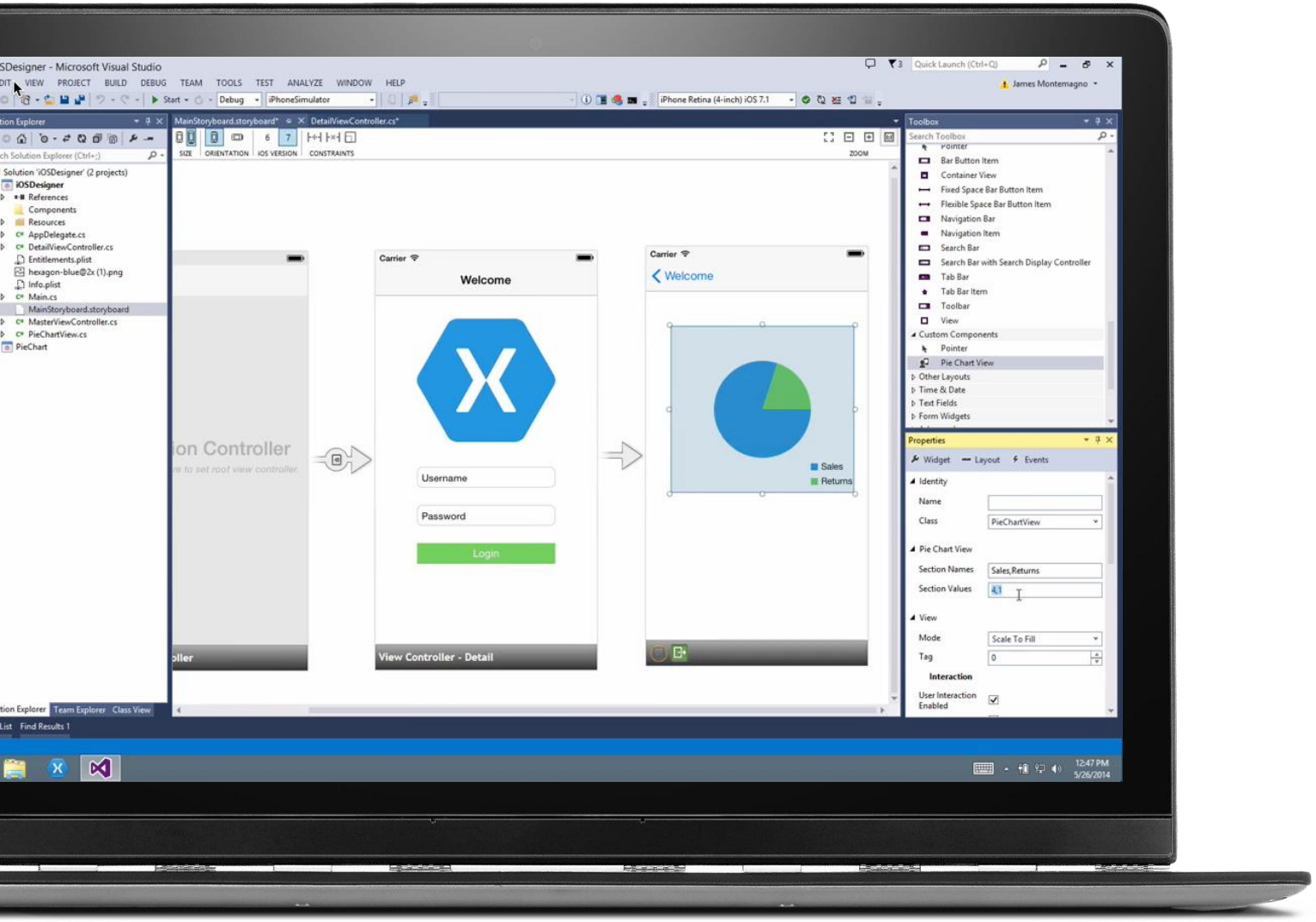


Code Sharing Stats

	iCircuit	Touch Draw
Android	14% / 86%	28% / 72%
iOS	30% / 70%	39% / 61%
Mac	12% / 88%	24% / 76%
Windows Phone	10% / 90%	✕

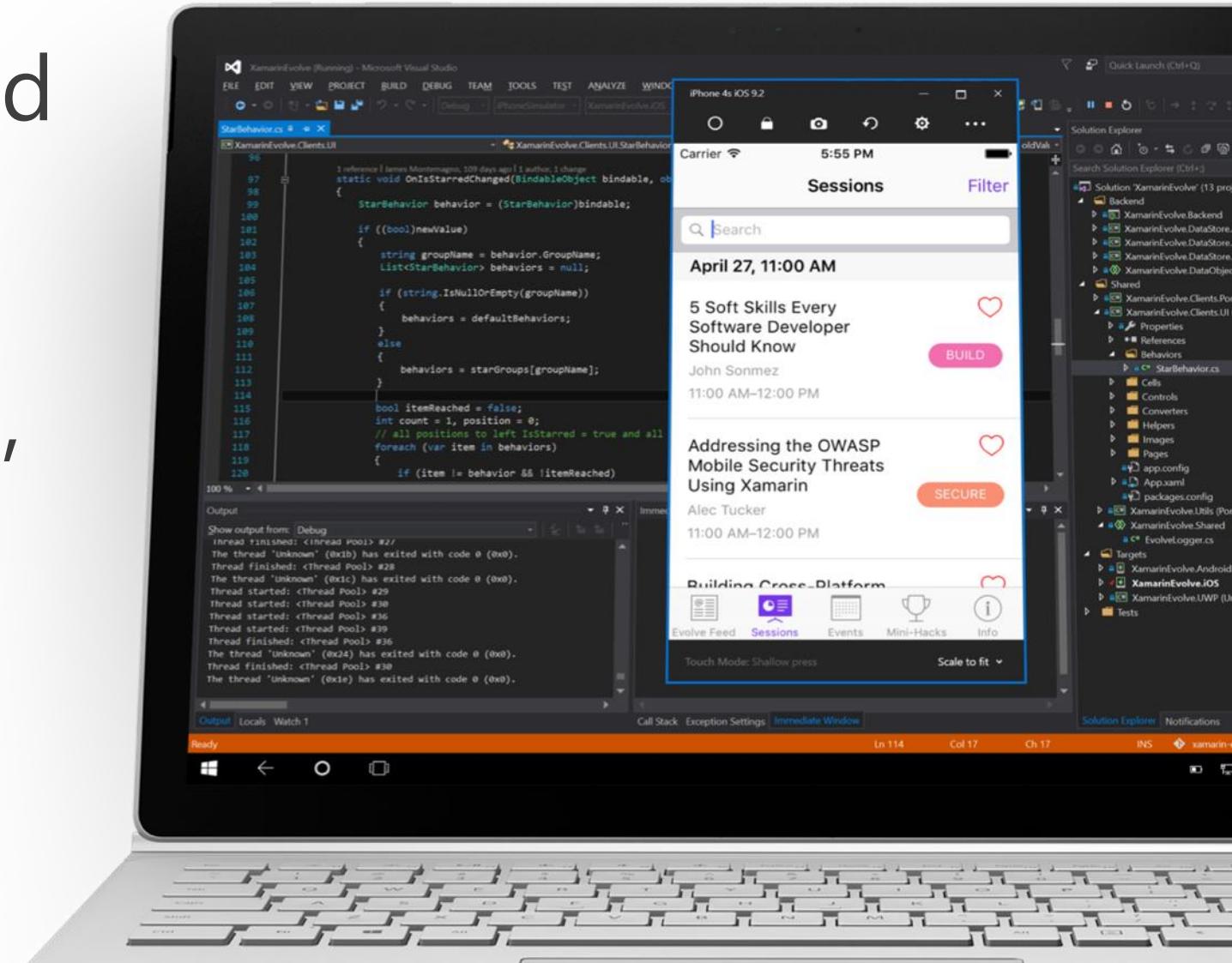
Development Experience

Visual Studio Integration



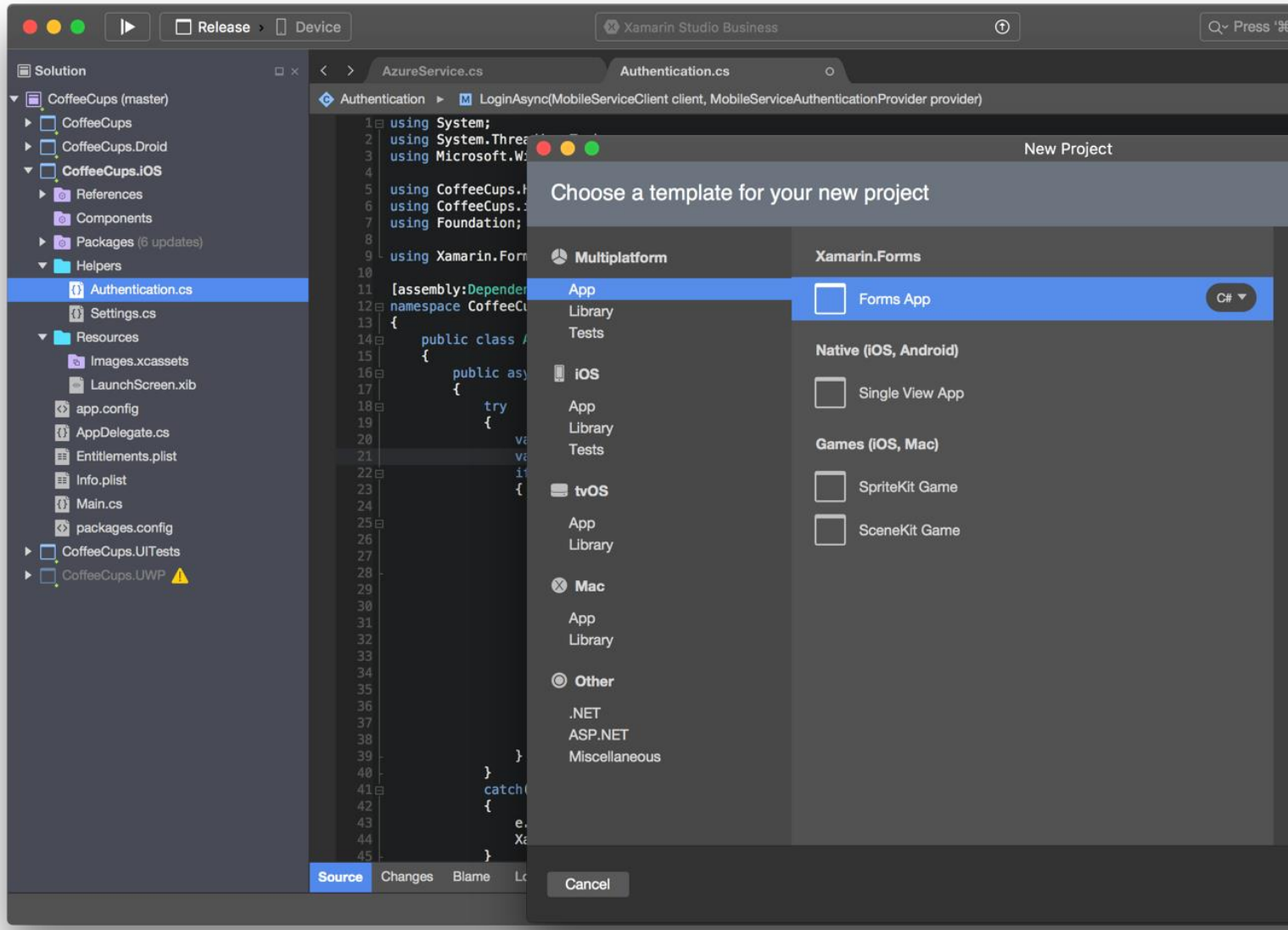
Visual Studio iOS Simulator Remoting

- Multi-Touch Enabled
- Pressure Sensitive
- Super Fast
- Rotate, screenshots, location changes
- Never leave VS



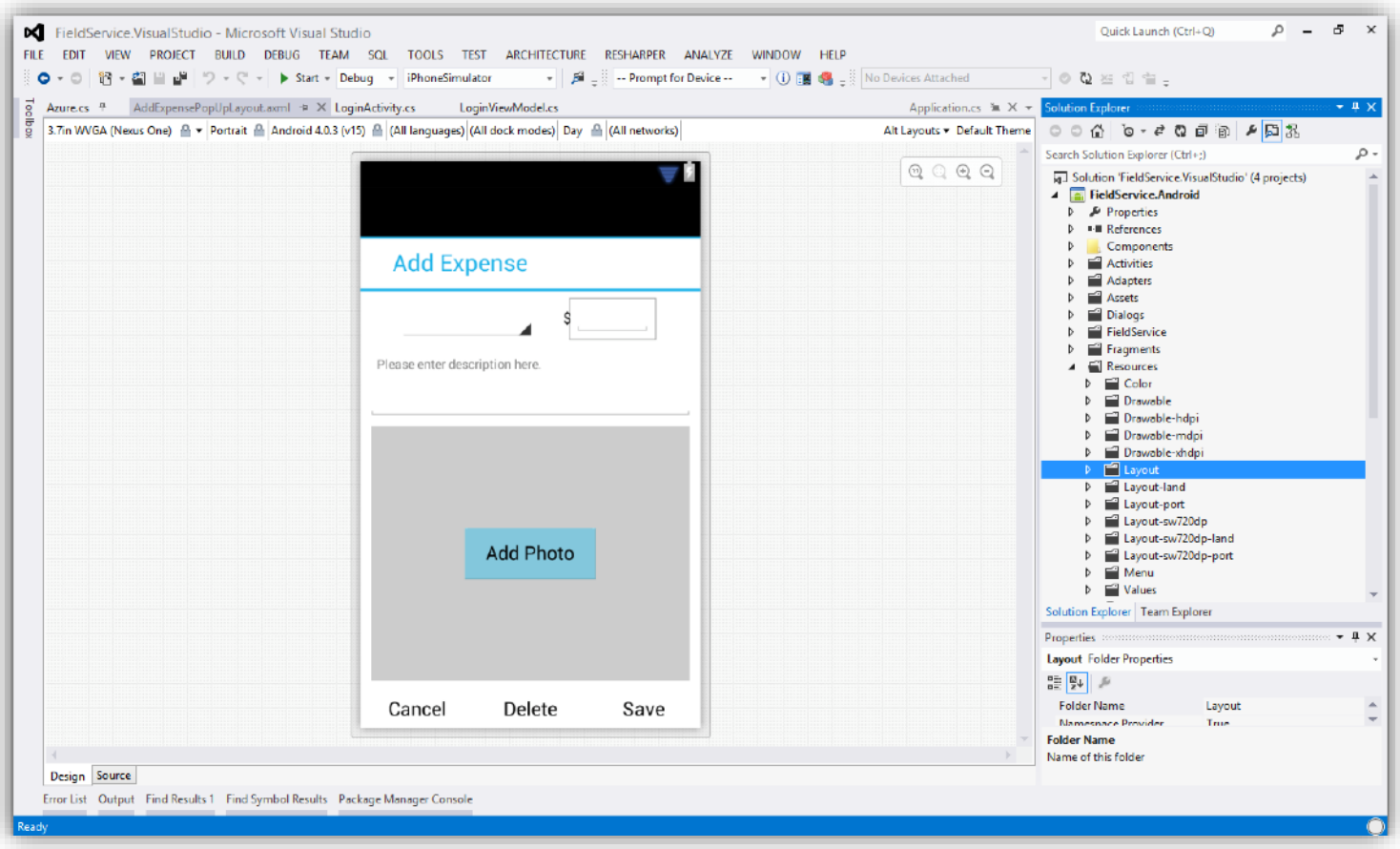
Visual Studio iOS USB Remoting

Xamarin Studio – Mac

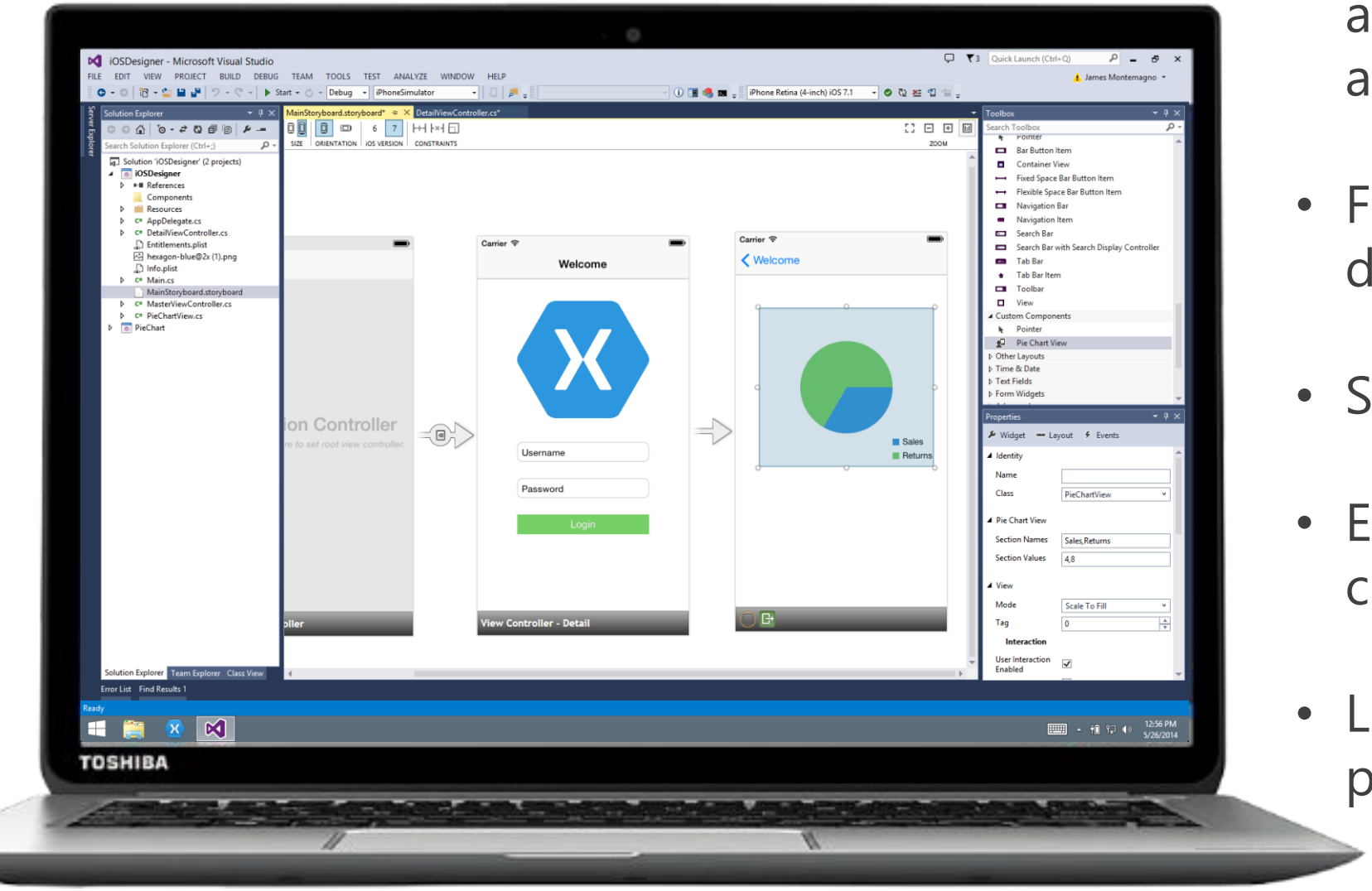


Android Designer

- Available in:
  - Xamarin Studio
  - Visual Studio
- Create UI with drag & drop simplicity
- Target multiple screen sizes, resolutions and Android versions
- Layouts saved in standard Android XML files

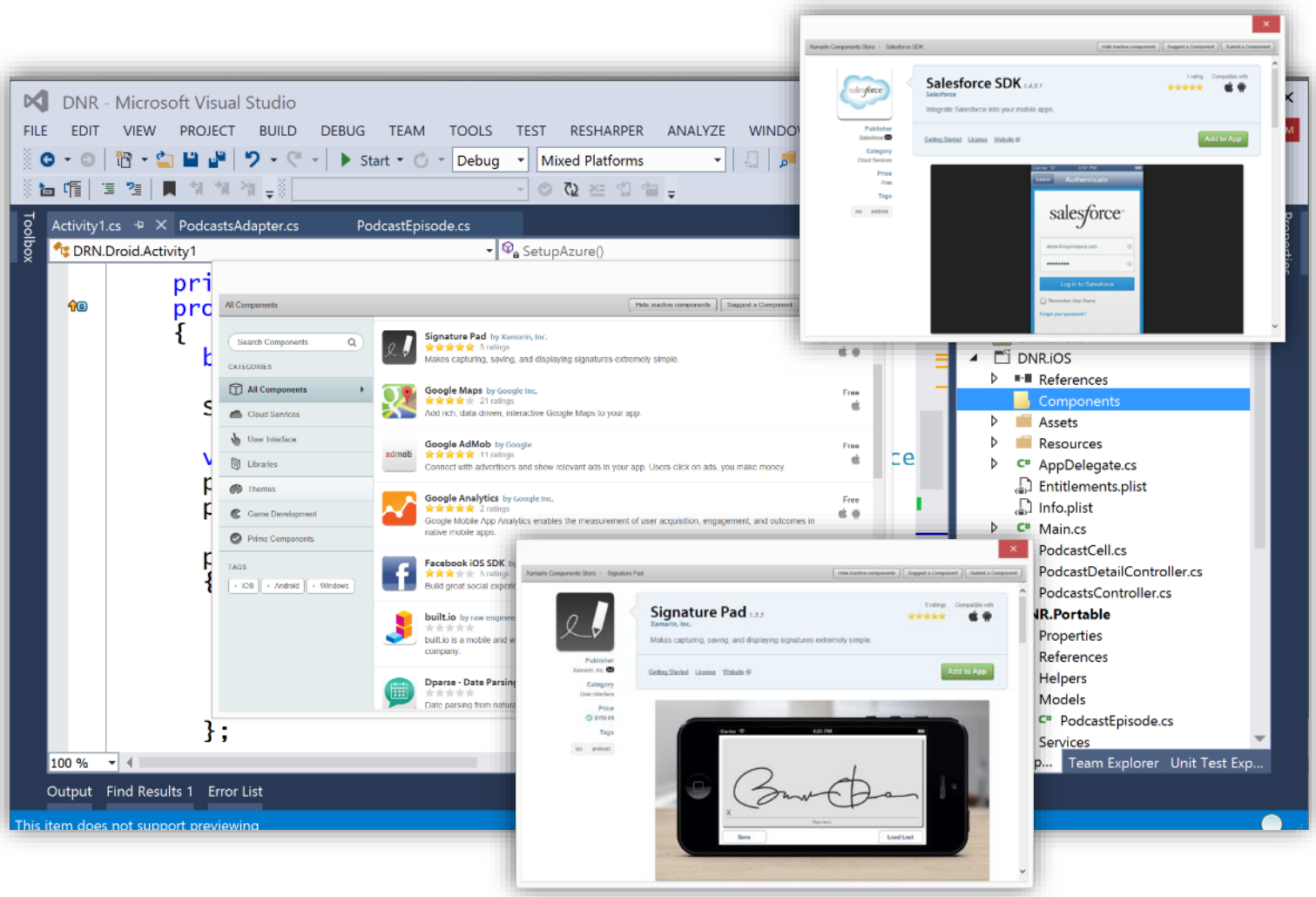


Xamarin Designer for iOS



- World's first iOS Designer available in Xamarin Studio and Visual Studio
- Follows familiar Visual Studio designer idioms
- Supports all UIKit elements
- Edit custom and 3rd party components
- Live preview of changes to properties

Xamarin Component Store

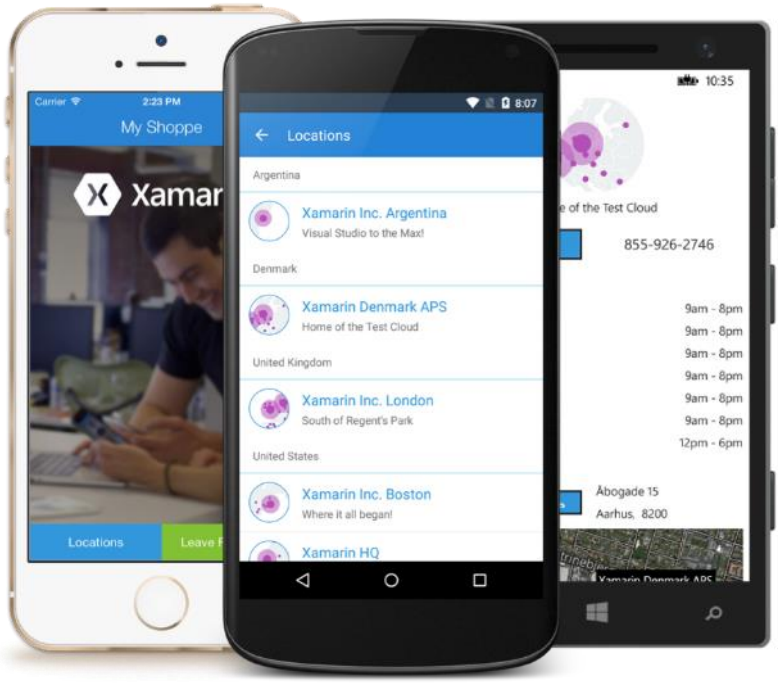


Build Apps Faster

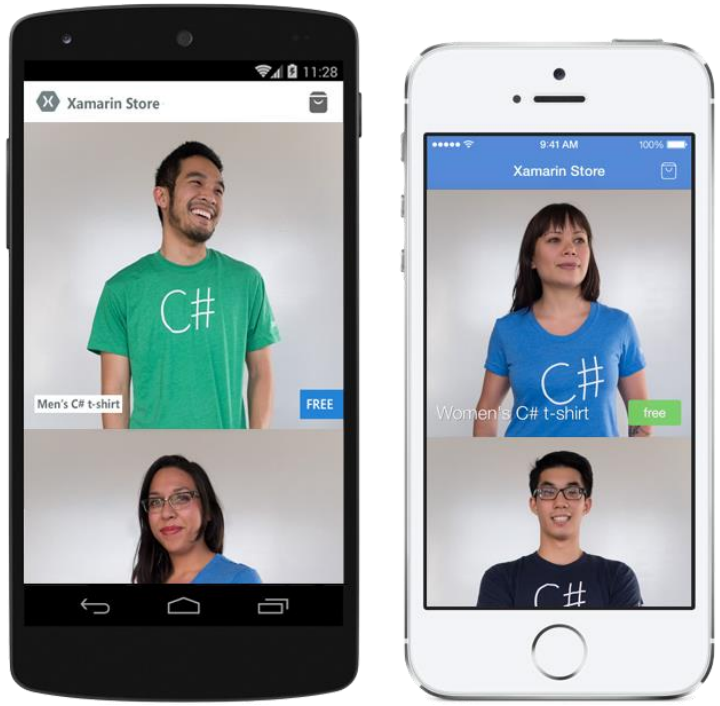
- Add high quality pre-built app components directly from Visual Studio and Xamarin Studio
- Beautiful, cross-platform UI controls, cloud services, and enterprise backend integrations are just a few clicks away

Pre-built App Templates

- `Xamarin.com/prebuilt`



My Shoppe
Connect with consumers

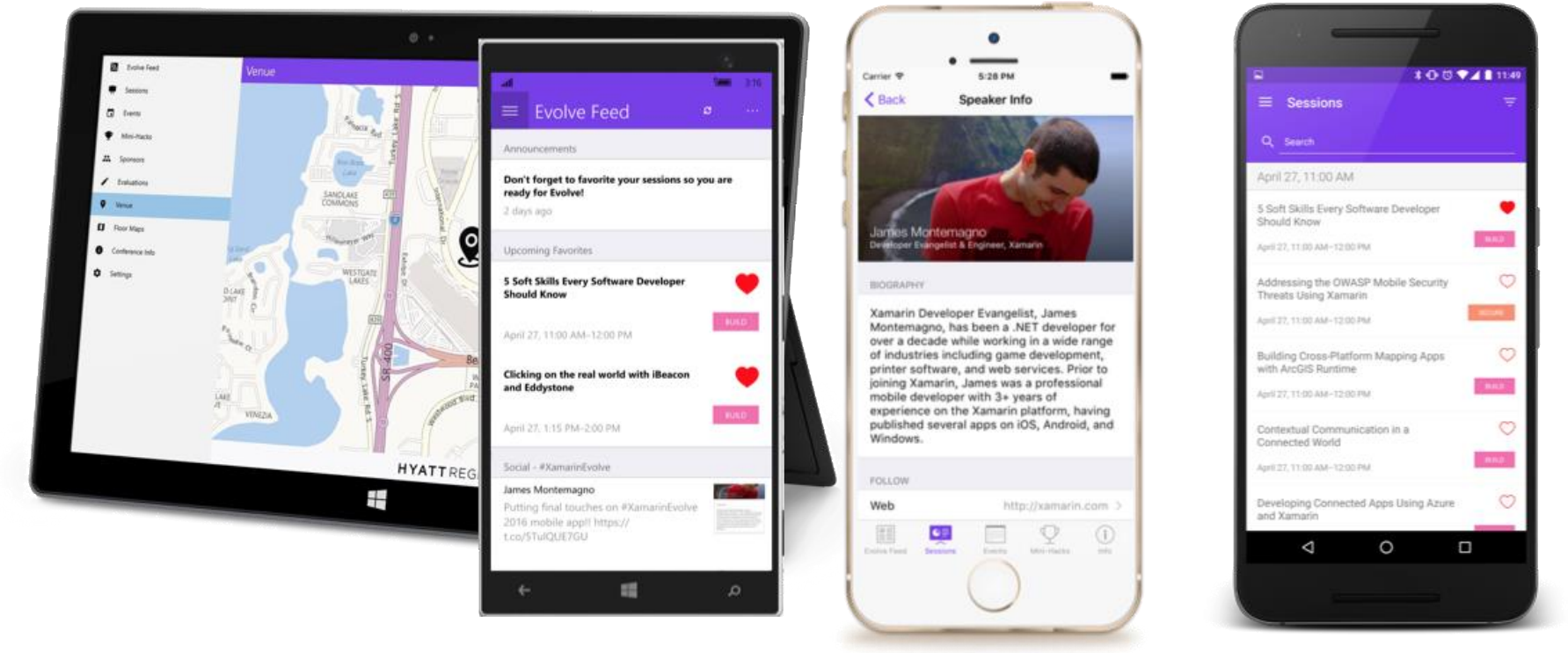


Store Application

Moments
Photo Sharing App

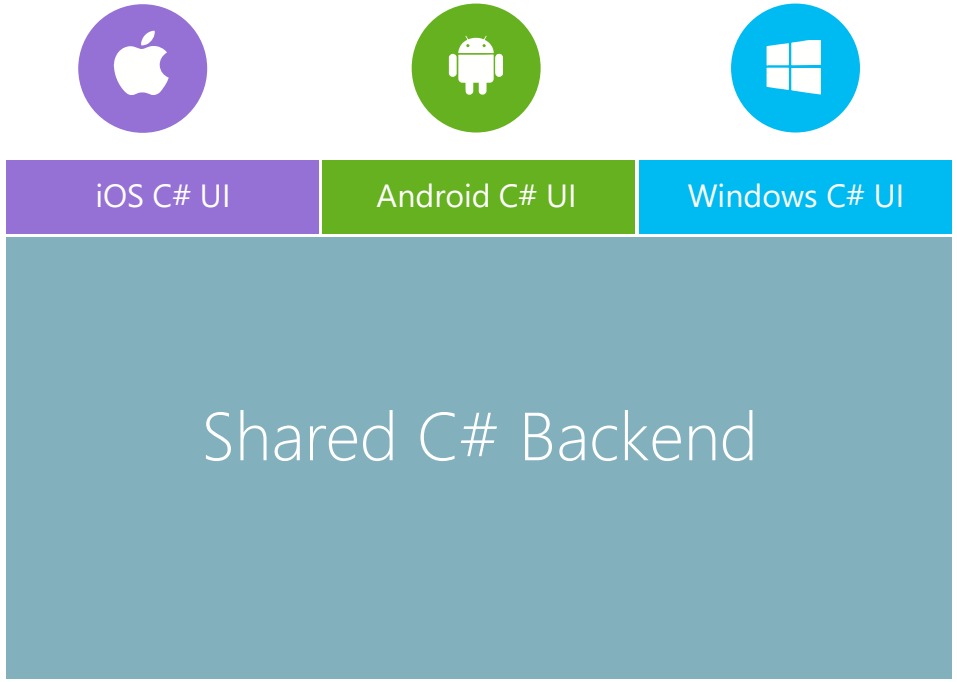
Let’s take a look

Meet Xamarin.Forms

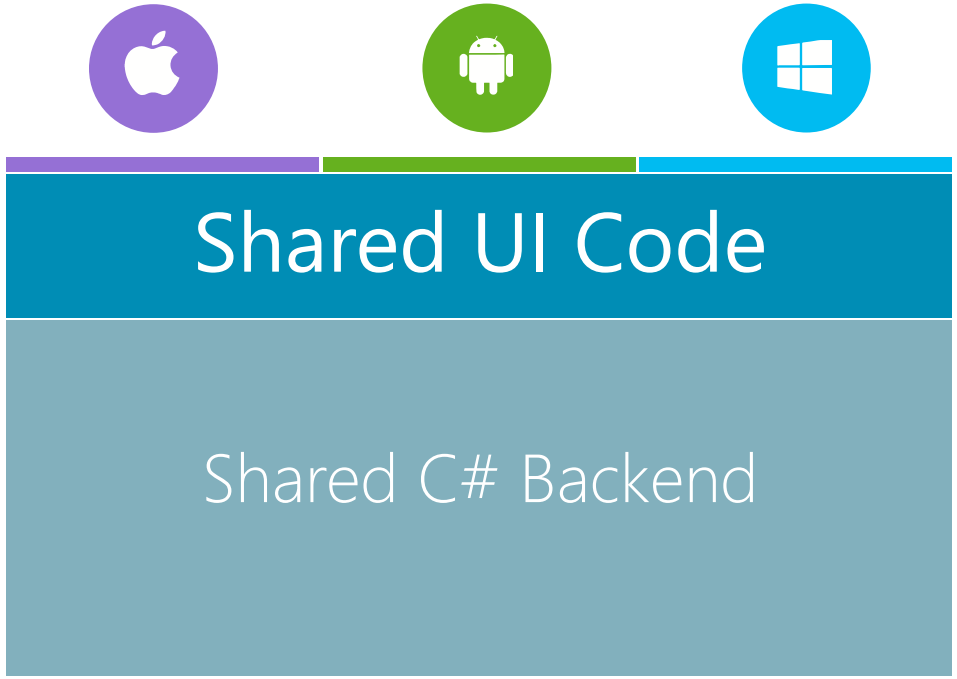


Build native UIs for iOS, Android, and Windows Phone from a single, shared C# codebase.

Xamarin + Xamarin.Forms

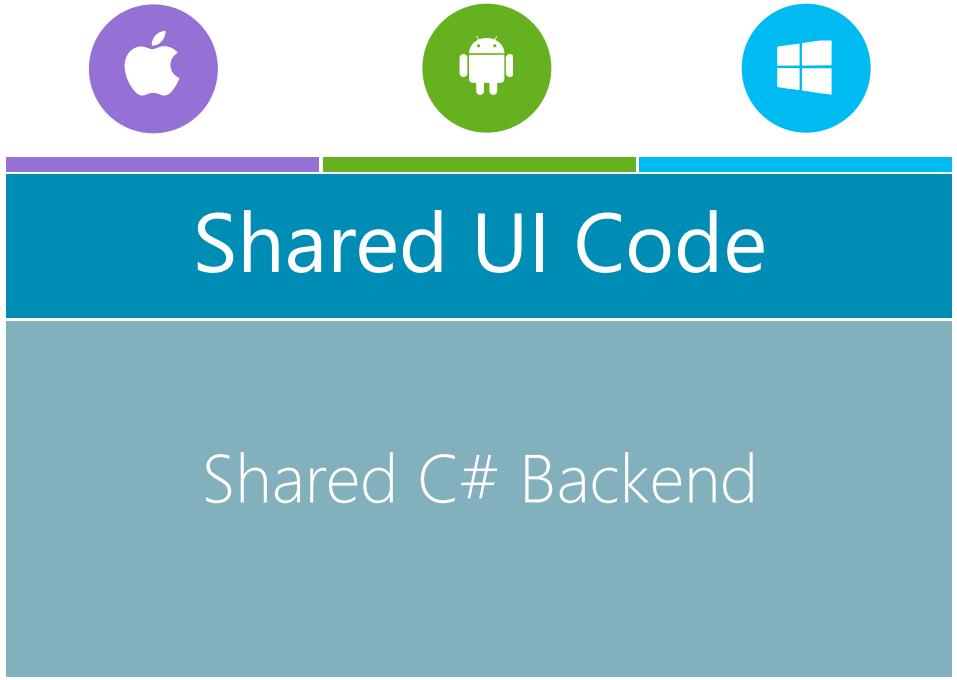


Traditional Xamarin Approach



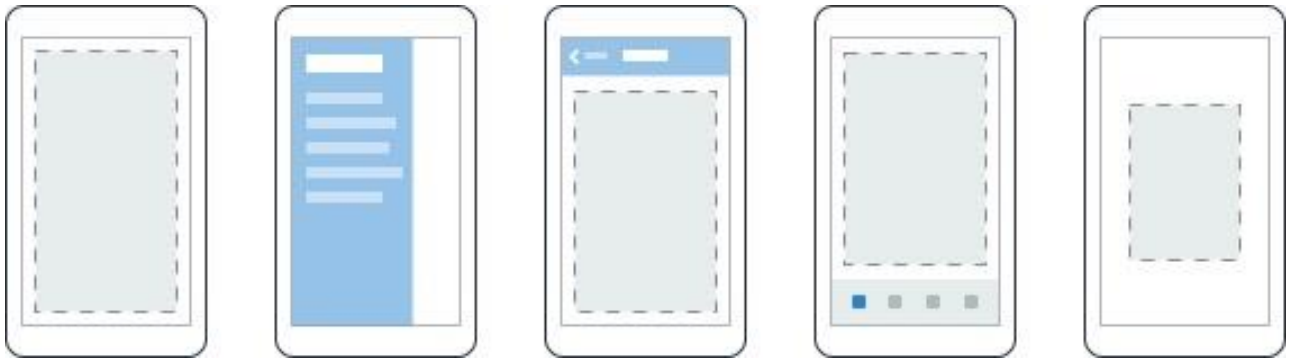
With Xamarin.Forms:
More code-sharing, all native

What's included



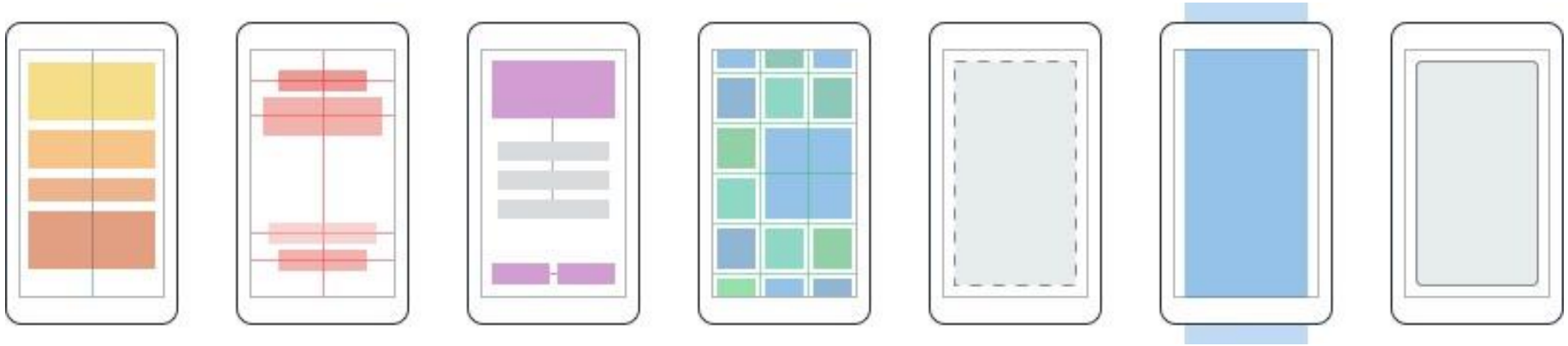
- ✓ 40+ Pages, layouts, and controls
  (Build from code behind or XAML)
- ✓ Two-way data binding
- ✓ Navigation
- ✓ Animation API
- ✓ Dependency Service
- ✓ Messaging Center

Pages



Content MasterDetail Navigation Tabbed Carousel

Layouts



Stack Absolute Relative Grid ContentView ScrollView Frame

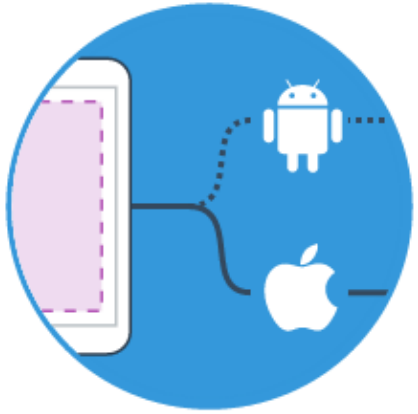
Controls

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

Extensibility

Embed Custom Views Anywhere

Not only are Xamarin.Forms pages mixable with custom screens, you can embed custom views built directly against Xamarin.iOS and Xamarin.Android into Xamarin.Forms pages.



Call Platform APIs via Shared Services

Need platform-specific functionality within a Xamarin.Forms page, such as querying the accelerometer? We've included services that quickly and easily abstract platform-specific API calls to keep you writing as much shared code as possible.

Easy migration to the Xamarin Platform

When your native platform integration requirements outgrow Xamarin.Forms, your shared code is easily migrated to Xamarin.iOS and Xamarin.Android using the full Xamarin platform.

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

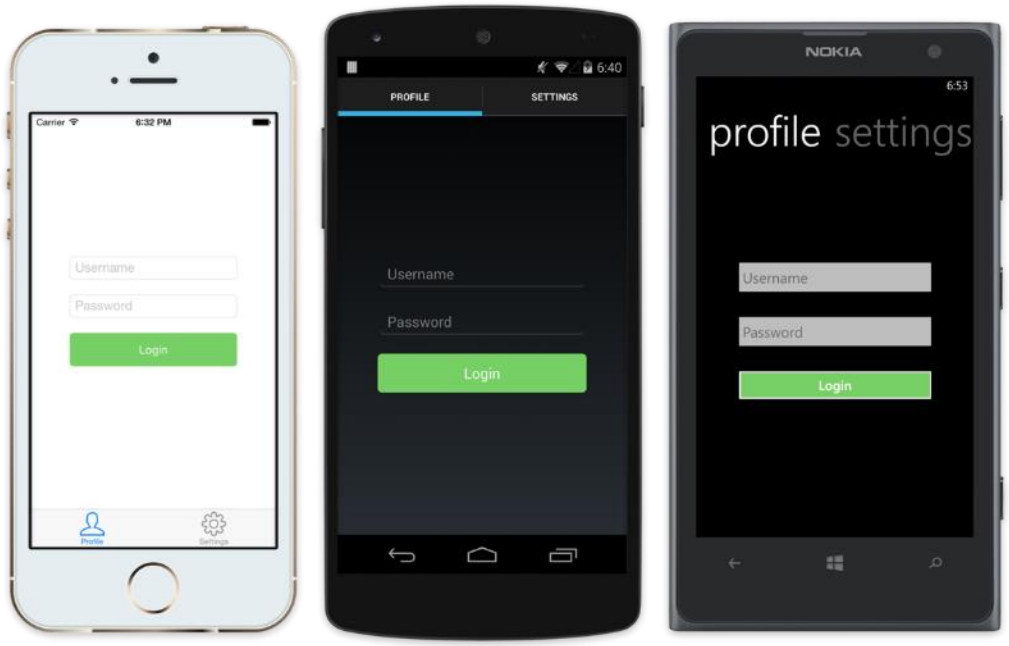
Windows	Xamarin.Forms
DataContext	BindingContext
{Binding Property}	{Binding Property}
ItemsSource	ItemsSource
ItemTemplate	ItemTemplate
DataTemplate	DataTemplate

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

Platform Customization

```
<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>
```

Native UI from shared code




```xml
<?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>
```

Let’s Take a Look

What’s new?

XAML Previewer

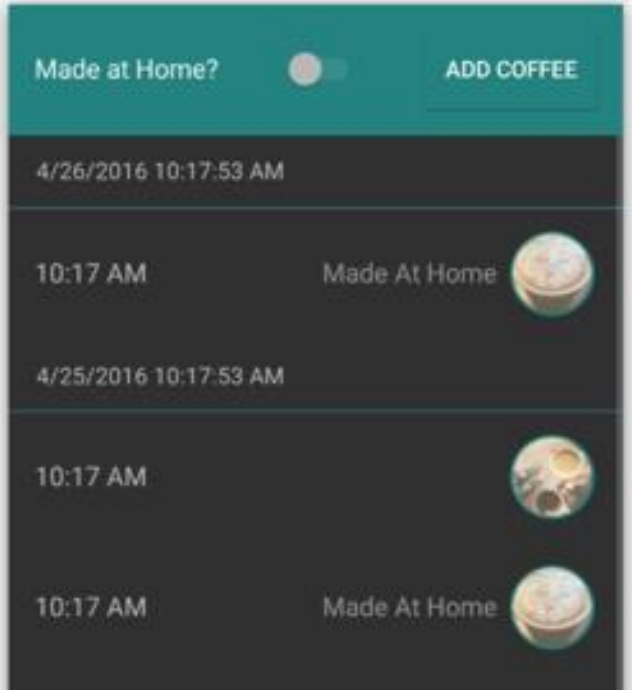
Debug | iPhoneSimulator › iPhone 6s iOS 9.3 | Xamarin Studio Business | Press '⌘.' to search

App.cs | ImageCircleRenderer.cs | ImageCircleRenderer.cs | CoffeesPage.xaml

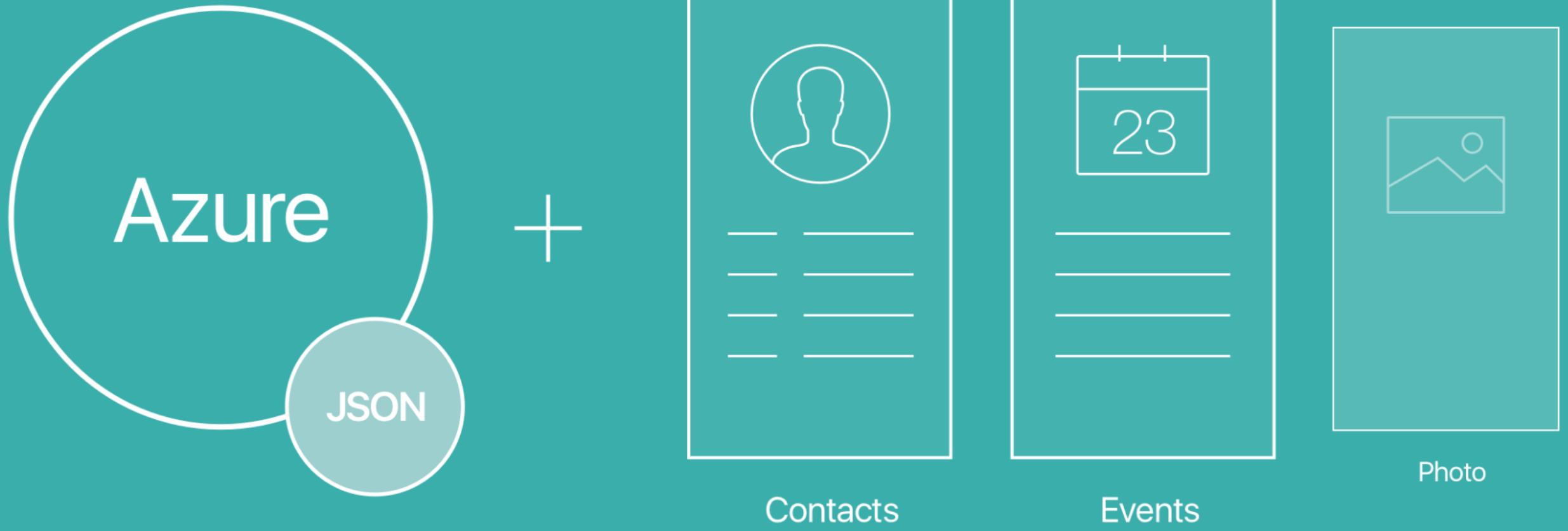
Hide Preview

```xml
<?xml version="1.0" encoding="UTF-8"?>
<ContentPage
    xmlns="http://xamarin.com/schemas/2014/forms"
    xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
    xmlns:local="clr-namespace:CoffeeCups;assembly=CoffeeCups"
    xmlns:controls="clr-namespace:ImageCircle.Forms.Plugin.Abstractions;assembl
    x:Class="CoffeeCups.CoffeesPage"
    Title="Cups Of Coffee">
     <AbsoluteLayout HorizontalOptions="FillAndExpand" VerticalOptions="FillAnd
        <Grid AbsoluteLayout.LayoutFlags="All"
              AbsoluteLayout.LayoutBounds="0,0,1,1"
              RowSpacing="0">
            <Grid.RowDefinitions>
                <RowDefinition Height="Auto"/>
                <RowDefinition Height="*"/>
                <RowDefinition Height="Auto"/>
            </Grid.RowDefinitions>
            <StackLayout Orientation="Horizontal" Padding="16" Spacing="16" Bacl
                <Label Text="Made at Home?"
                       TextColor="White"
                       VerticalOptions="Center"
                       HorizontalOptions="Start"
                       FontSize="16">
```

Device: Phone Tablet Platform: Android iOS

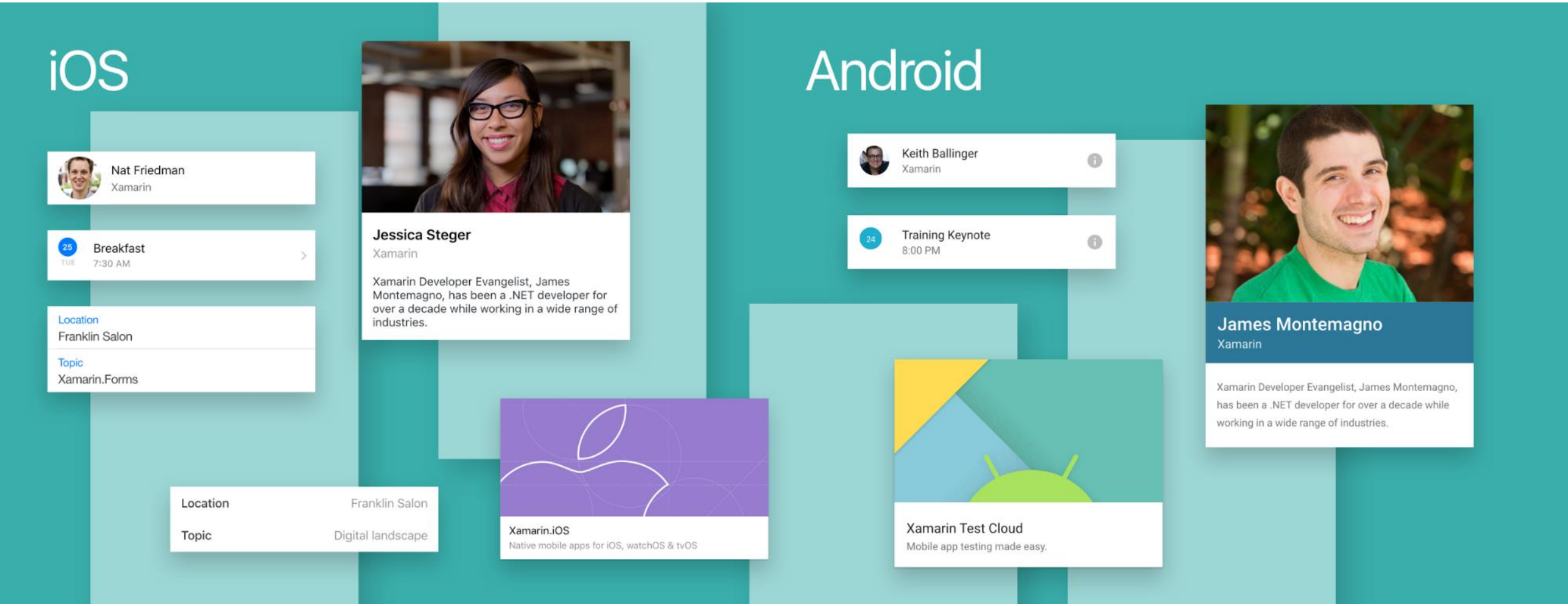


DataPages



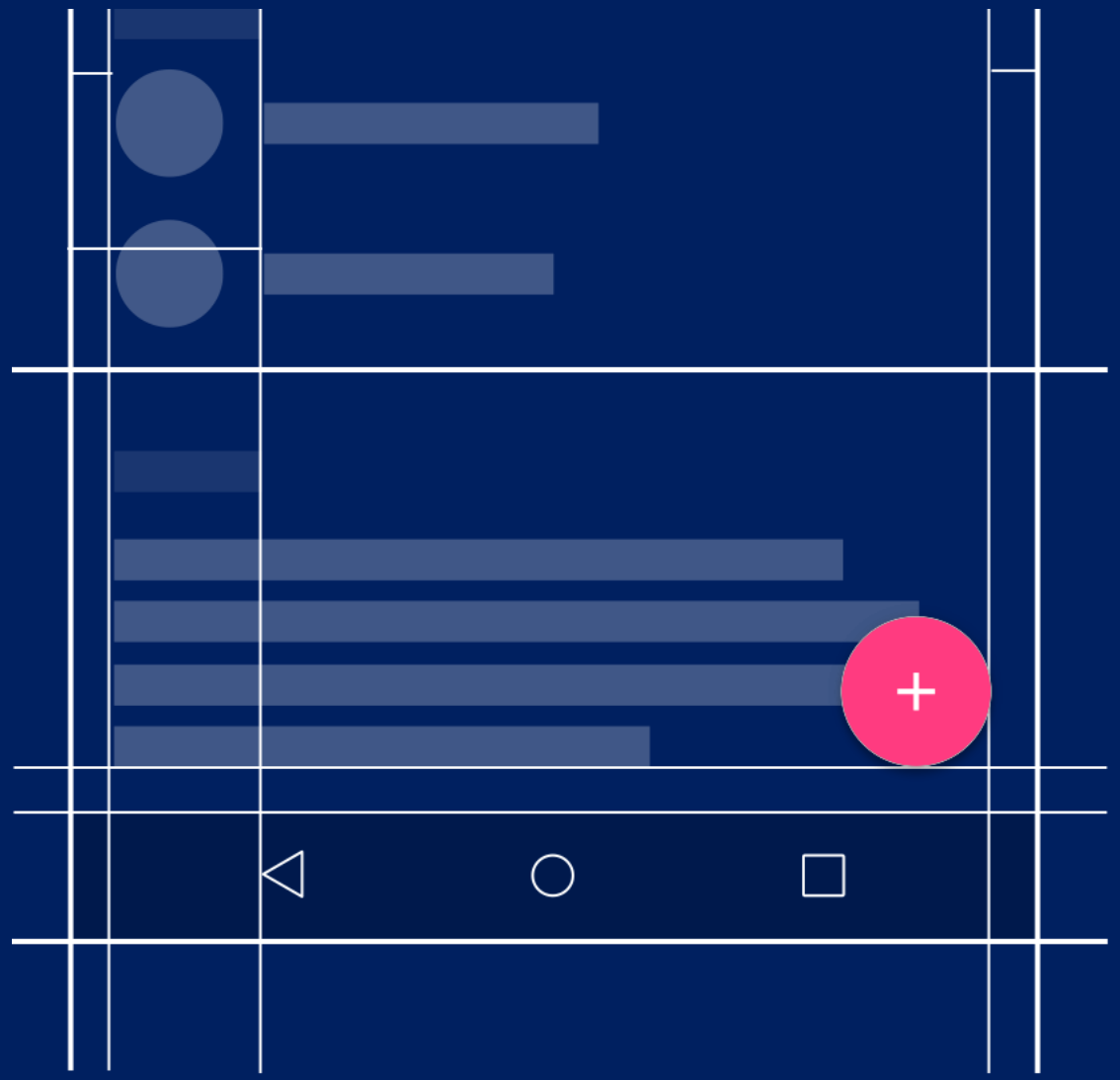
https://developer.xamarin.com/guides/xamarin-forms/datapages/

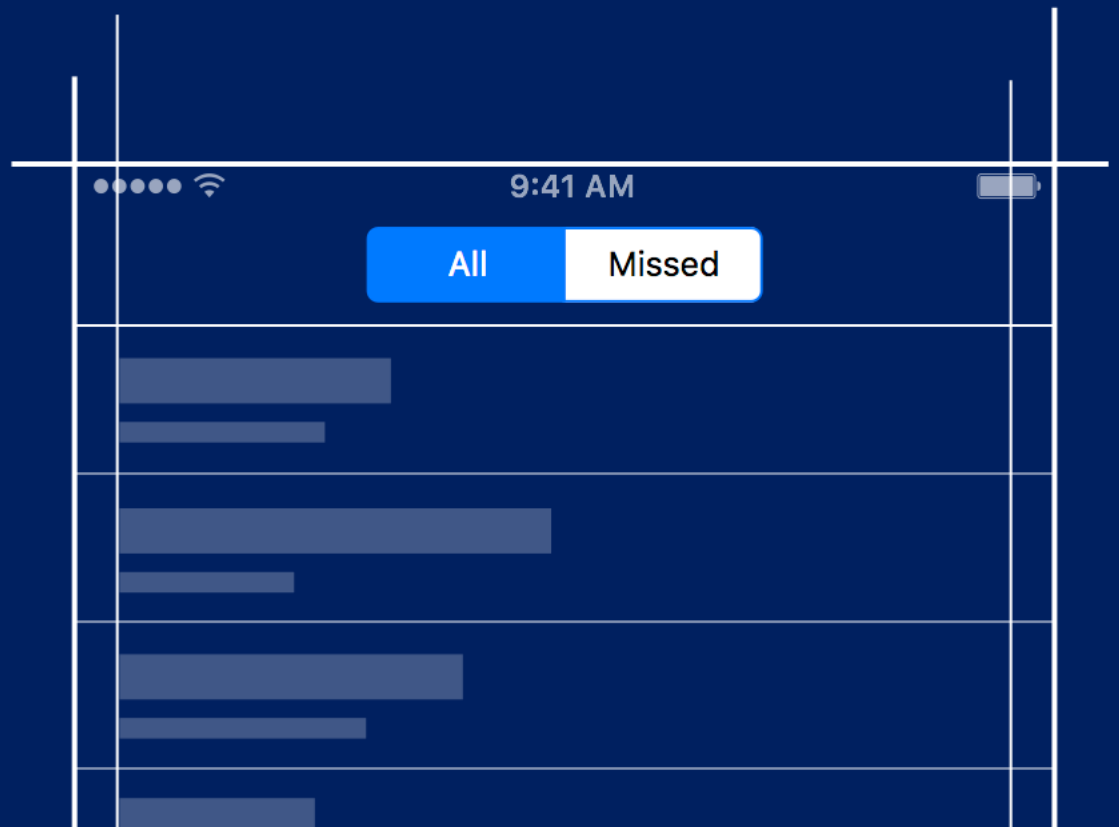
Themes



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

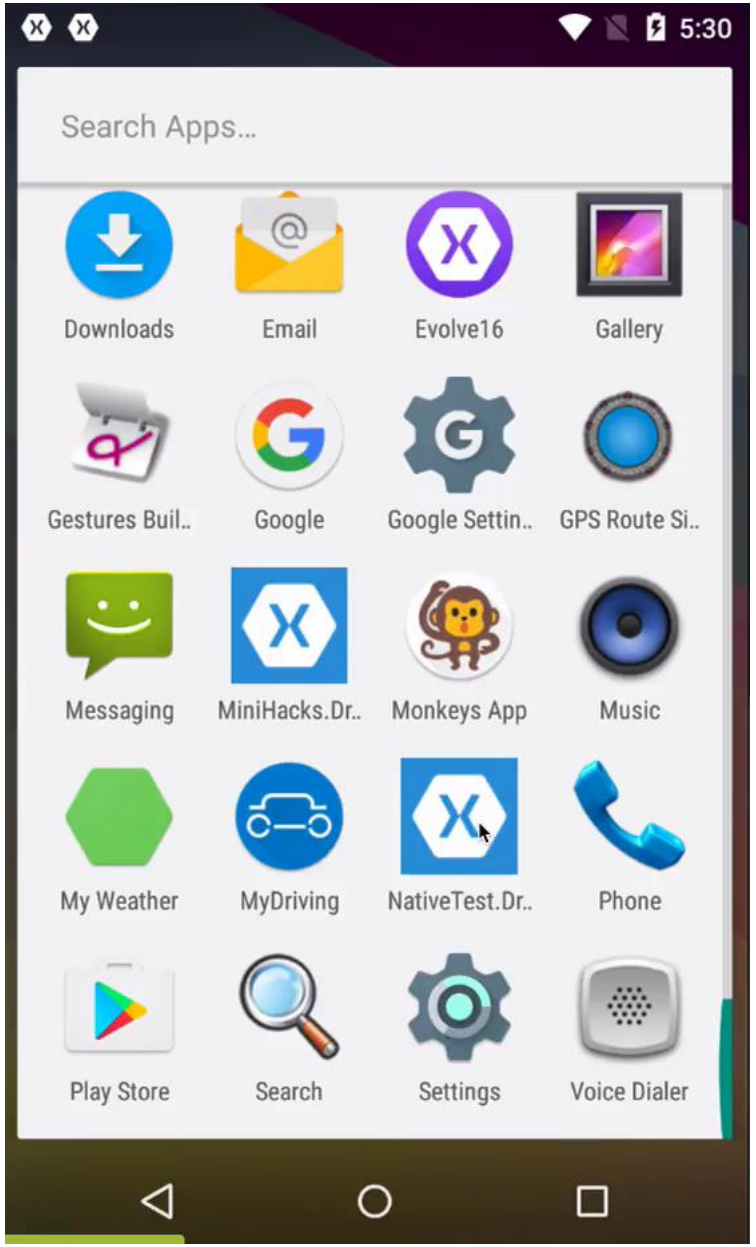
Native Embedding



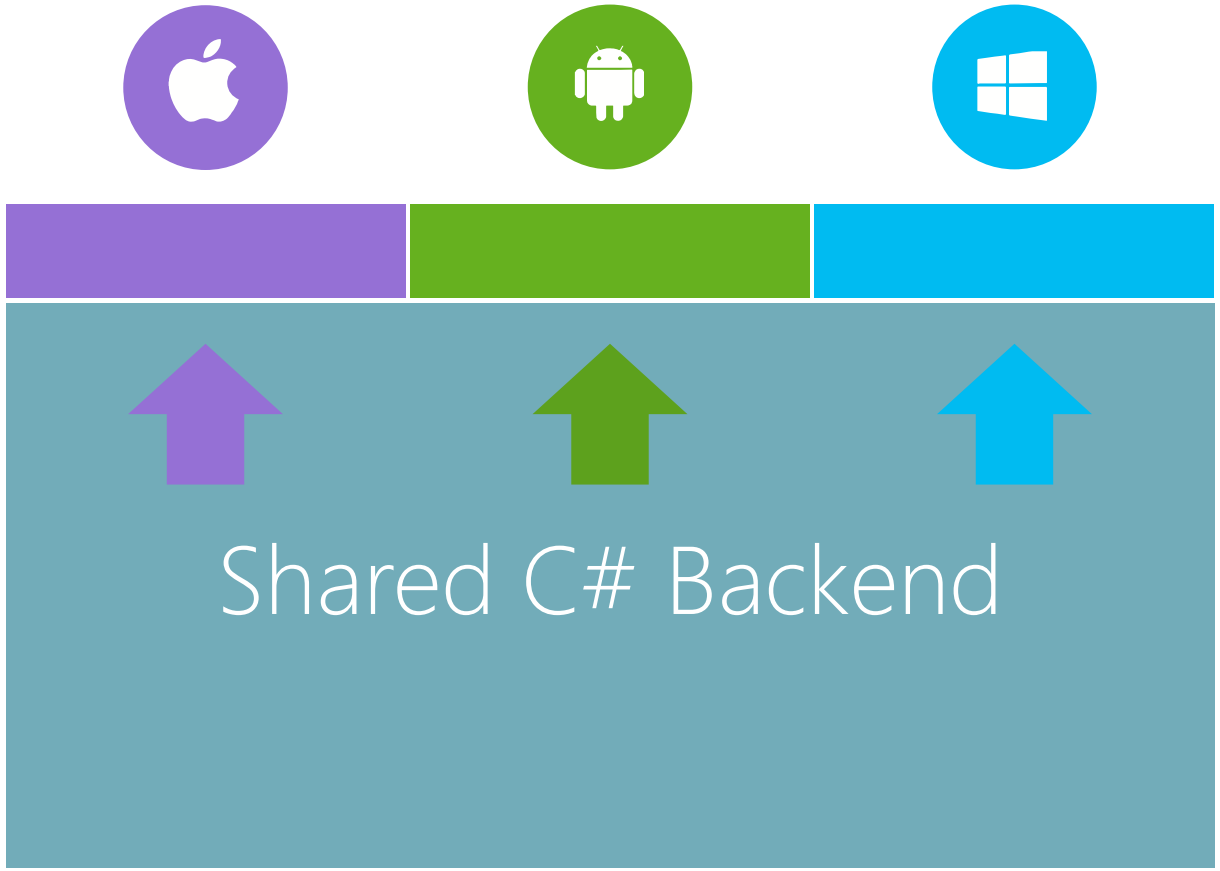


Native Embedding

```
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
```



Plugins for Xamarin

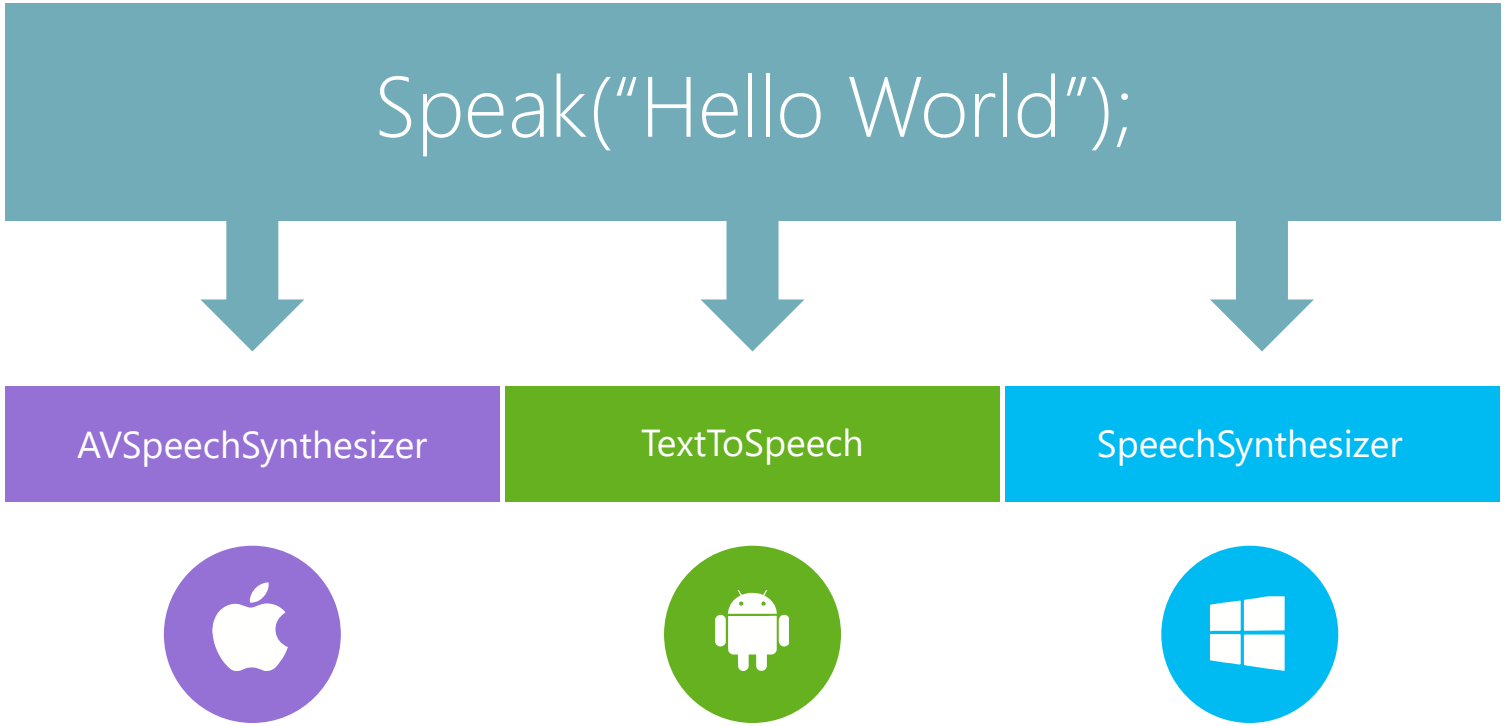

Shared C# Backend

Platform Specific Code

UI+APIs	UI + APIs	UI + APIs
Battery	Battery	Battery
GPS	GPS	GPS
Lights	Lights	Lights
Notifications	Notifications	Notifications
Settings	Settings	Settings
Text To Speech	Text To Speech	Text To Speech

What if we didn't have to write this code?

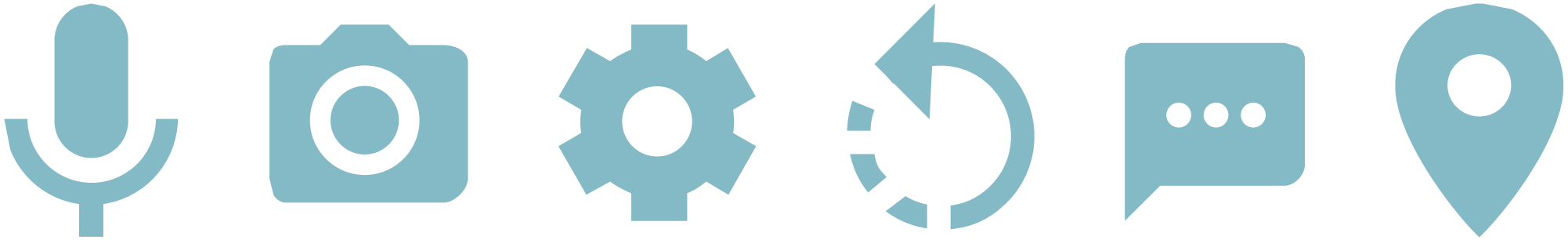
What if we could access it from shared code?


Speak("Hello World");
AVSpeechSynthesizer
TextToSpeech
SpeechSynthesizer

Plugins for Xamarin

github.com/xamarin/plugins

Common API



Let’s Take a Look

Test

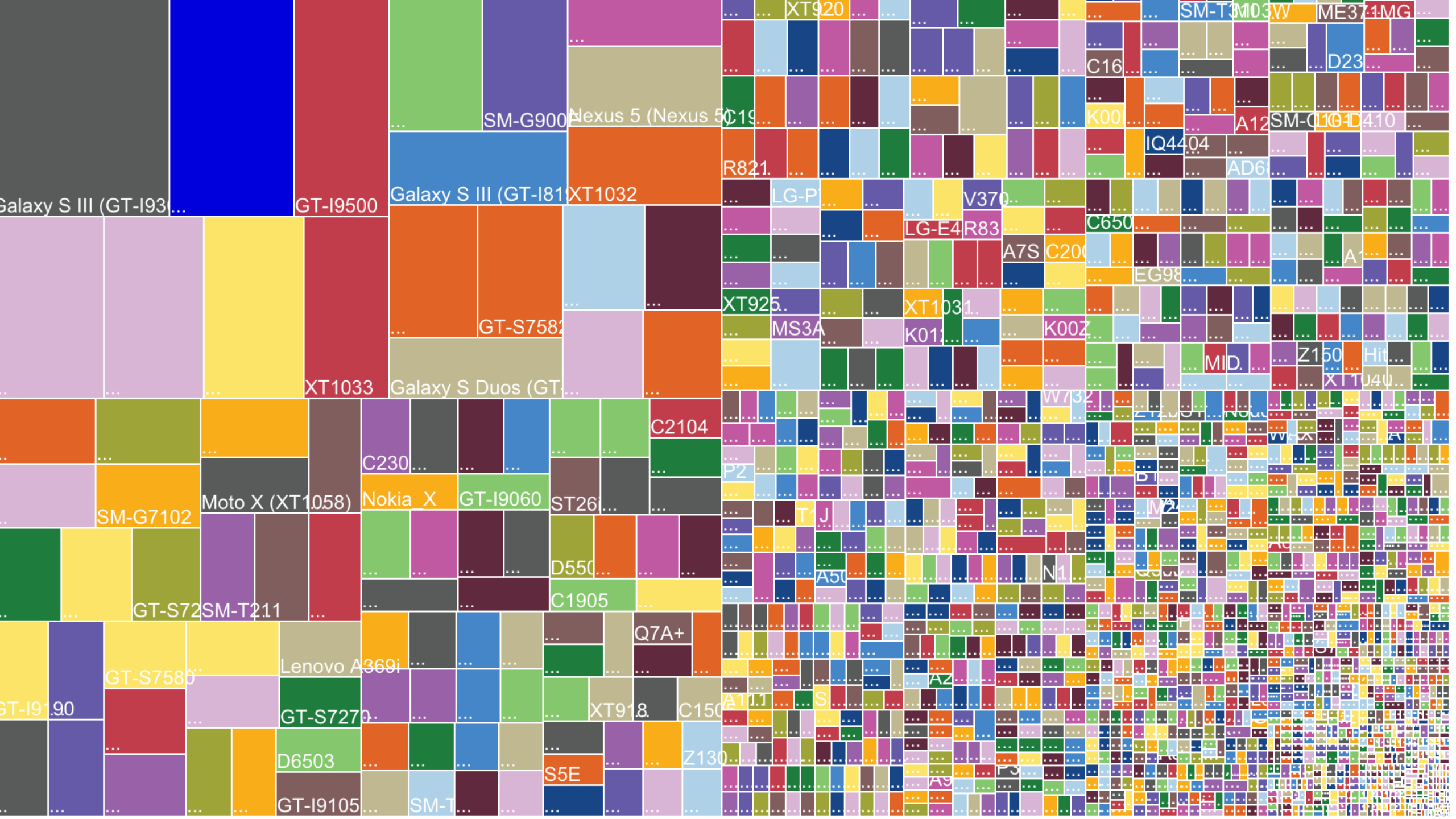
Challenges

- Device Fragmentation
- App Complexity
- Fast Release Cycles
- Short Sessions
- High Mobile User Expectations

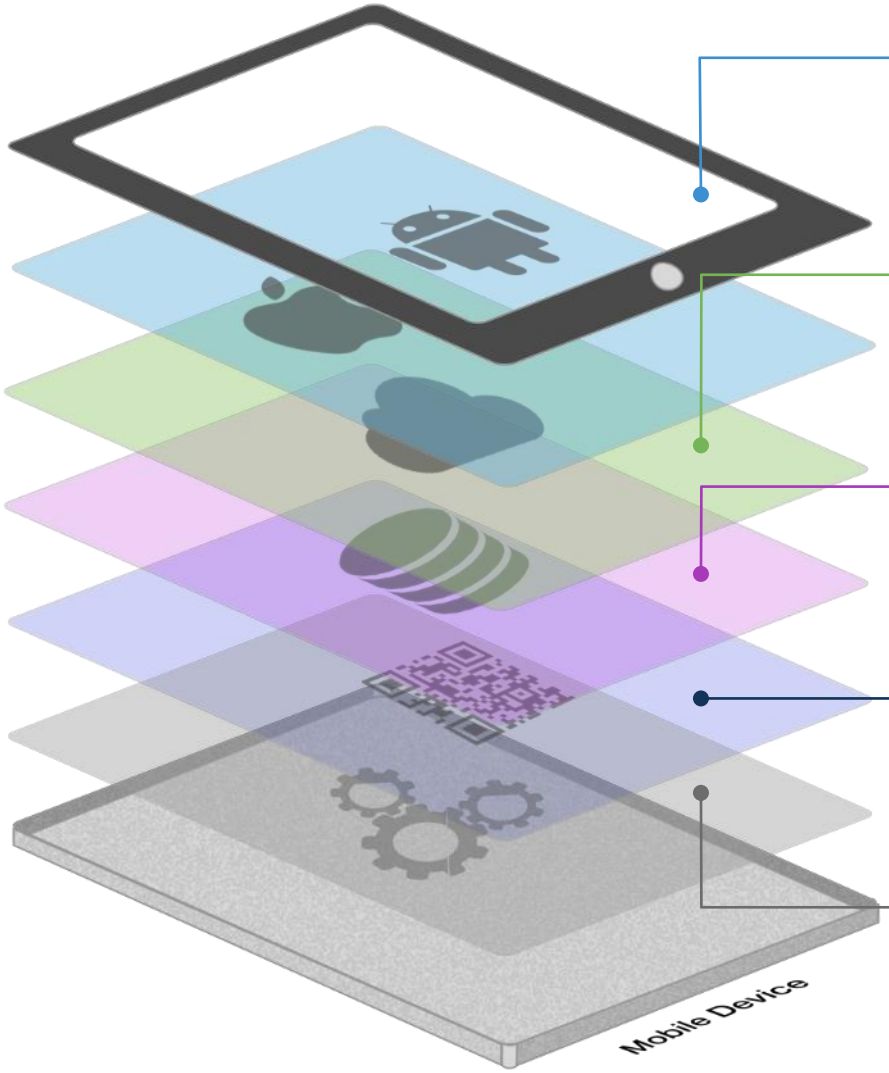
On Mobile, Quality is Hard

- 5 OS versions
- 20 Devices
- 20 Languages
- 35 Locales
- 6 Screen sizes

- 9 OS versions
- 19K+ Distinct devices
- 39 Languages
- 57 Locales
- 27 Screen sizes
- 15 Manufacturers
- 6 Screen configurations


Galaxy S III (GT-I93
GT-I9500
SM-G900
Nexus 5 (Nexus 5)
Galaxy S III (GT-I81
XT1032
GT-S7582
XT1033
Galaxy S Duos (GT
C2104
C230
Moto X (XT1058)
Nokia X
GT-I9060
SM-G7102
GT-S72
SM-T211
D550
C1905
Q7A+
Lenovo A369i
GT-S7580
GT-I9190
GT-S7270
D6503
GT-I9105
XT918
C150
S5E
Z130
XT920
C16
K00
A12
IQ4404
AD6
D23
R821
LG-P
V370
LG-E4
R83
A7S
C650
EG98
XT925
MS3A
XT1031
K01
K00Z
MID
Z150
W732
P2
T1
J
A50
N1
A2
A1
S
A3
P3

App Complexity



Thousands of iOS and Android APIs

Public web APIs

Backend integrations

Third party libraries

Memory, CPU and network constraints

Tap

Scroll

Swipe

Pinch

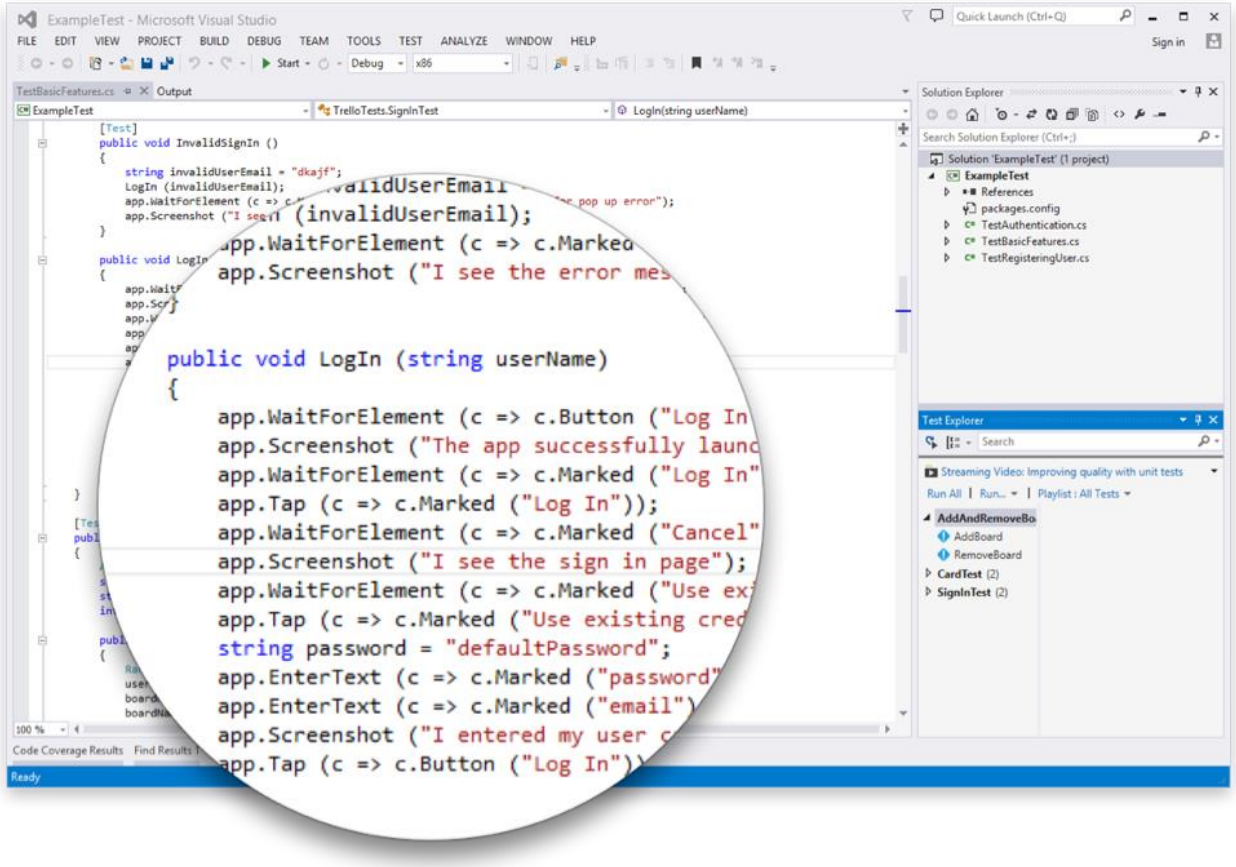
Multi Finger

Text Entry

Rotation

GPS

Introducing Xamarin.UITest



Create Automated User Interface tests all in C#

Upload to the Test Cloud or run against a Device or Simulator

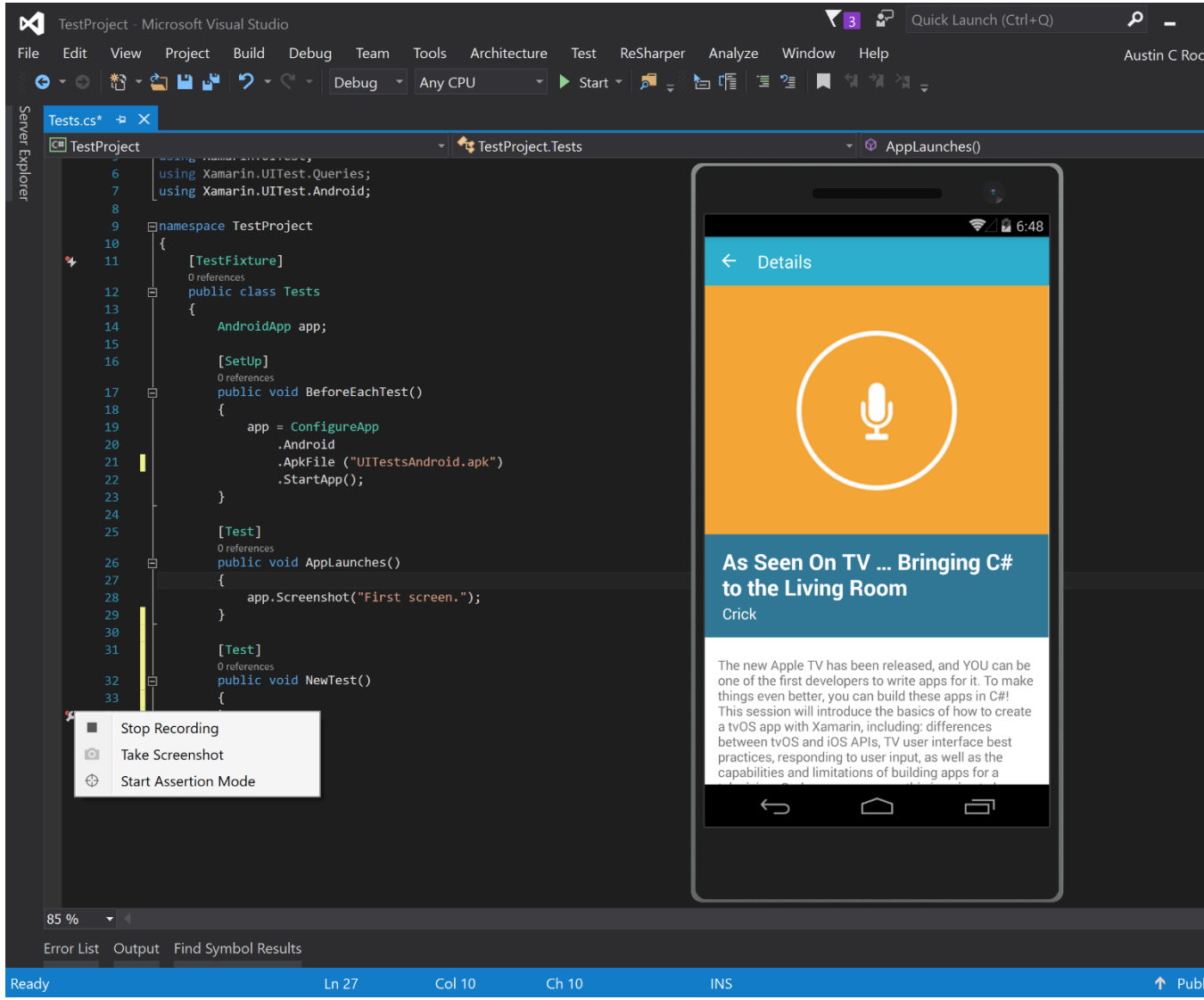
Run directly from Visual Studio or Xamarin Studio

Freely available for testing on a Simulator*

Works on **ANY** app: Native, Hybrid, or Xamarin

Test Recorder for Visual Studio

- Record UITests from Visual Studio
- Replay or ship to Test Cloud
- Use Touch Screen on Android Emulator



Test Recorder for Mac

- Record UITests for iOS and Android apps
- Replay or ship to Test Cloud

Xamarin Test Recorder

Record Run Stop Output Export

Open Device: Nexus 5 (Lollipop) - 5.1 App: Simple Credit Car...

Tap EditText [creditCardNumberText]

Type text EditText "12345... 123456789012345

Tap Button "Validate Credit Card"

Tap TextView "Credit card numbe..."

```
public void NewTest ()
{
  app.Tap(x => x.Class("EditText").Id("creditCardNumberText"));
  app.EnterText(x => x.Class("EditText").Id("creditCardNumberText"),
  "123456789012345");
  app.Tap(x => x.Class("Button").Id("validateButton").Text("Validate
  Credit Card"));
  app.Tap(x =>
  x.Class("TextView").Id("errorMessagesText").Text("Credit card number
}
```

Xamarin Android Player - [Nexus 5 (Lollipop)]

12:59

Enter Credit Card Number

123456789012345

Validate Credit Card

Credit card number is too short.

Xamarin Test Cloud

xamarin.com/testcloud

Flipboard > Sep 23, 7:09

https://testcloud.xamarin.com/

Xamarin test cloud 〉 Flipboard 〉 master 〉 Sep 23, 2014 7:30:39 PM

New Test Run | Vinicius

Overview

ALL RESULTS

Sign in

User creates an account 5

- Given I am on the start screen
- When I go to the login screen 5
- And I enter valid credentials
- Then I should be logged in

User signs in with Facebook ✓

User signs in with Google ✓

User has incorrect password ✓

User has incorrect email 3

User signs out ✓

Reading articles

User reads the cover story ✓

User reads the News section ✓

User reads the Technology section ✓

User reads Twitter articles ✓

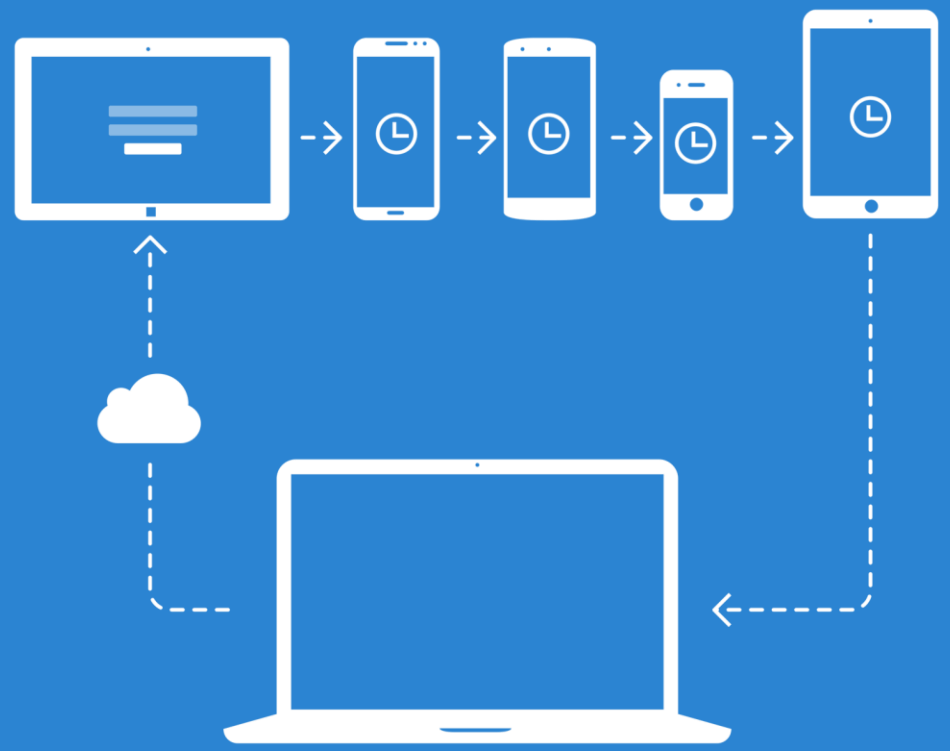
User adds a section ✓

User comments on an article 3

Filter devices

LG Nexus 5 Android 4.4.2	Samsung Galaxy S II Android 4.1.2	Samsung Galaxy S III Android 4.1.2	Samsung Galaxy S Duos Android 4.0.4	Samsung Galaxy Core Android 4.1.2
Samsung Galaxy Grand Duos Android 4.2.2	Samsung Galaxy S Duos 2 Android 4.2.2	LG Nexus 4 Android 4.4.2	HTC One Android 4.4.2	Samsung Galaxy Note Android 4.1.2

Approaches to Mobile Testing



Device Remoting

Tests are performed one at the time, which consumes more time and delays bugs detection

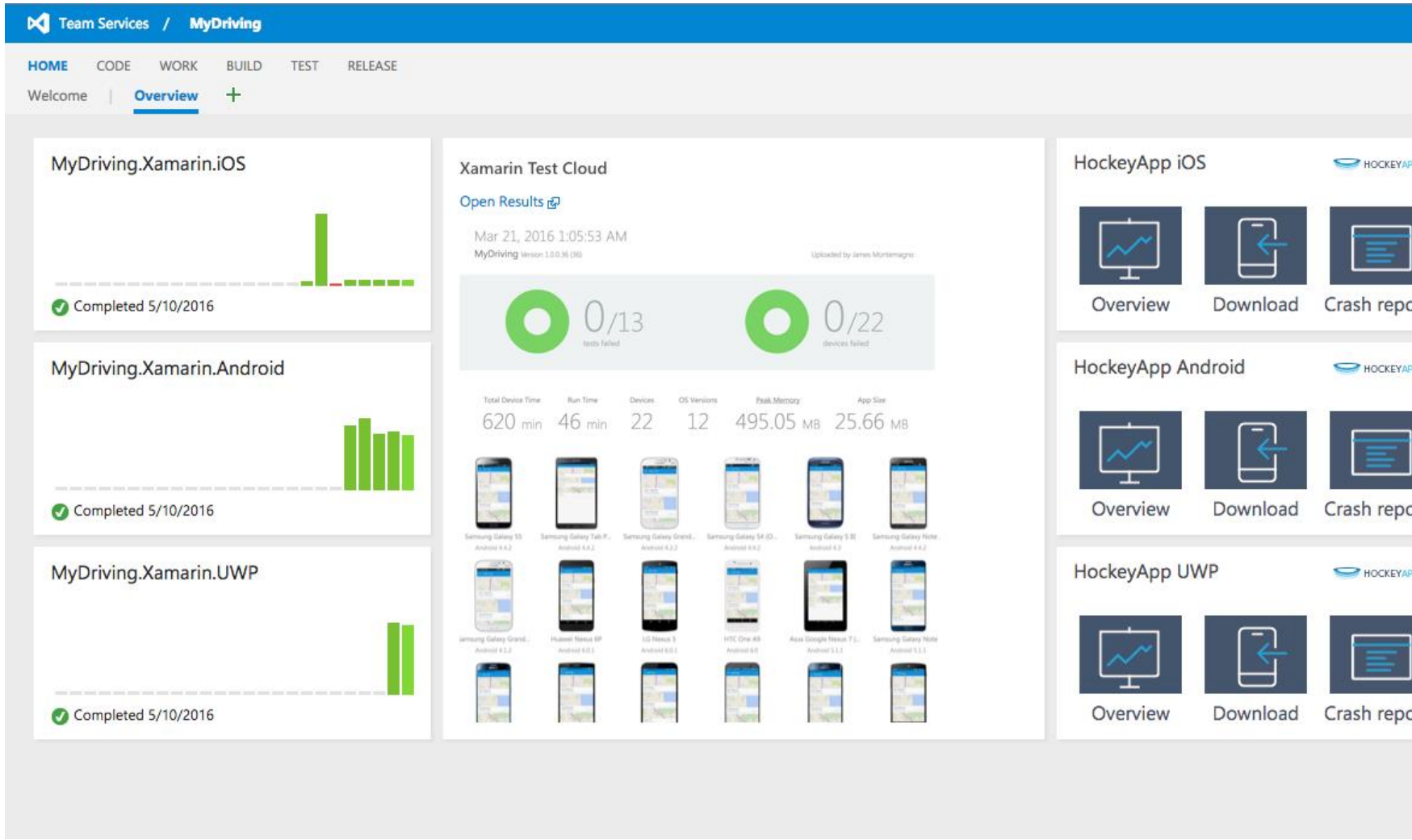
Automated Testing

Test on thousands of devices simultaneously, saving lots of time and detecting bugs more quickly

Build

Visual Studio Team Services

- Code
- Work
- Build
- Test
- Release



Visual Studio Team Services - Work

Team Services / MyDriving

James Montemagno

HOME CODE WORK BUILD TEST RELEASE

Search work items

Backlogs Queries

You can now add more backlog levels to this team. Access this setting through the settings dialog.

Features

Backlog items

Current

Sprint 1

Future

Sprint 2

Sprint 3

Sprint 4

Sprint 5

Sprint 6

Backlog items

Backlog Board

Forecast Off Mapping Off Parents Hide In progress items Show

New Create query Column options

Filter

Type Product Backlog Item

Title Add

Order	Work Item Type	Title	State	Effort	Value Area	Iteration Path	Tags
1	Bug	Fix NullReferenceException in MyTrips.Services.OBDDataProcessor+...	New		Business	MyDriving	
2	Product Backlog...	Use Release Management to deploy against multiple HockeyApp co...	New		Business	MyDriving	
3	Product Backlog...	Update Xamarin Test Cloud for Android builds with correct devices ...	New		Business	MyDriving	
4	Bug	Fix IllegalArgumentException in com.squareup.picasso.Picasso.load,...	New		Business	MyDriving	
5	Bug	Fix crash in MyTrips.Droid.Services.GeolocationHelper.get_LocationS...	New		Business	MyDriving	
6	Bug	Fix ArgumentException in ObdLibAndroid.ObdWrapper+<Disconne...	New		Business	MyDriving	
7	Bug	Fix IllegalArgumentException in android.view.WindowManagerGlob...	New		Business	MyDriving	
8	Bug	Fix IllegalArgumentException in com.squareup.picasso.Picasso.load,...	New		Business	MyDriving	
9	Bug	Fix SQLiteException in Microsoft.WindowsAzure.MobileServices.SQL...	New		Business	MyDriving	
10	Bug	Fix NotImplementedException in Microsoft.Azure.Devices.Client.Dev...	New		Business	MyDriving	
11	Bug	Fix NullReferenceException in MyTrips.Services.OBDDataProcessor+...	New		Business	MyDriving	
12	Bug	Fix NullReferenceException in ObdLibAndroid.ObdWrapper+<Init>...	New		Business	MyDriving	
13	Bug	Fix IllegalArgumentException in md511b398f68db4d0ef3494f1d176...	New		Business	MyDriving	
14	Bug	Fix ActivityNotFoundException in android.app.Instrumentation.chec...	New		Business	MyDriving	
15	Bug	Fix NullPointerException in com.google.android.gms.common.inter...	New		Business	MyDriving	
16	Bug	Fix crash in Windows.Devices.Bluetooth.Rfcomm.RfcommDeviceServ...	New		Business	MyDriving	
17	Bug	Fix crash in Acr.UserDialogs.UserDialogsImpl.<>c__DisplayClass2_0...	New		Business	MyDriving	
18	Bug	Fix crash in System.Runtime.ExceptionServices.ExceptionDispatchInf...	New		Business	MyDriving	
19	Bug	Fix crash in Windows.UI.Xaml.Controls.ContentDialog.ShowAsync	New		Business	MyDriving	
20	Bug	Fix (Exception in Windows.Networking.Sockets.StreamSocket.Cance...	New		Business	MyDriving	

Visual Studio Team Services - Build

Team Services / MyDriving

James Montemagno

HOME CODE WORK BUILD TEST RELEASE

Explorer

Search work items

Search all definitions...

- My favorites
- Team favorites
  - MyDriving.Xamarin.Android — Completed 5 hours ago
  - MyDriving.Xamarin.iOS — Completed 5 hours ago
  - MyDriving.Xamarin.UWP — Completed 5 hours ago
- Build definitions
  - All build definitions
  - MyDriving.Services
  - MyDriving.Xamarin.Android
  - MyDriving.Xamarin.Android-Feature
  - MyDriving.Xamarin.Android-Regression
  - MyDriving.Xamarin.iOS
  - MyDriving.Xamarin.iOS Evolve
  - MyDriving.Xamarin.iOS-Feature
  - MyDriving.Xamarin.iOS-Regression
  - MyDriving.Xamarin.UWP
  - MyDriving.Xamarin.UWP-Feature
  - MyDriving.Xamarin.UWP-Regression
- XAML definitions
  - All XAML definitions

Definitions / MyDriving.Xamarin.Android-Feature | Builds

Build Options Repository ! Variables Triggers General Retention History

Save Queue build... Undo

Add build step...

- Replace tokens for BingMaps — *Replace Tokens*
- NuGet restore src/MobileApps/MyDriving.XS.sln — *NuGet Installer*
- Update Version Name — *Version Assemblies*
- Update Version Code — *Version Assemblies*
- Download keystore — *Command Line*
- Activate Xamarin license — *Xamarin License*
- Build and Sign Android Project — *Xamarin.Android*
- Deactivate Xamarin license — *Xamarin License*
- Build tests — *MSBuild*
- Test in Xamarin Test Cloud — *Xamarin Test Cloud*
- Copy Files to: $(build.artifactstagingdirectory) — *Copy Files*
- Publish Artifact: drop — *Publish Build Artifacts*

Replace tokens for BingMaps

Source Path: src/MobileApps/MyDriving/MyDriving.Utils/

Target File Pattern: Logger.cs

Advanced

Control Options

- Enabled [x]
- Continue on error [x]
- Always run [ ]

More Information

Visual Studio Team Services - Test

Team Services / MyDriving

James Montemagno

HOME CODE WORK BUILD TEST RELEASE

Test plan Runs Machines* Xamarin Test Cloud Load test

Recent Test Cloud test runs

- MyDriving
- MyDriving

Xamarin Test Cloud API Key

Save API Key

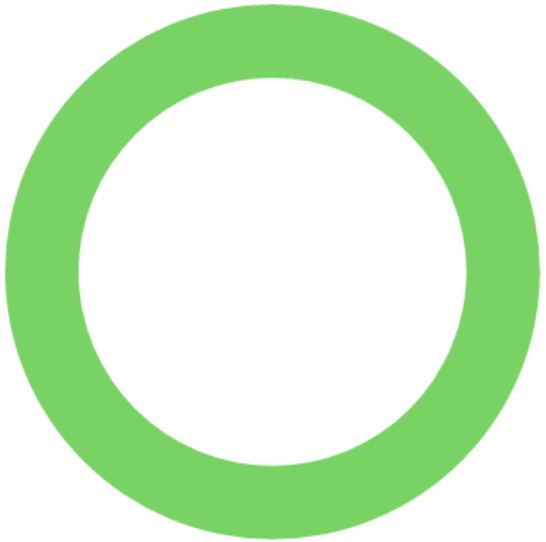
Help with Xamarin Test Cloud

MyDriving

2016-05-12T18:01:22.591Z

Tests

View Xamarin Test Cloud for detailed reports



0/12

tests failed

Devices

View Xamarin Test Cloud for detailed reports

0/19

devices failed

Current Trip Tests(iOS)

- Record Trip Test

Login Tests(iOS)

- Skip Authentication Test

12 / 12	Apple iPhone 6s - ios 9.0.2
12 / 12	Apple iPhone 6s Plus - ios 9.1
12 / 12	Apple iPhone 6s Plus - ios 9.1

Visual Studio Team Services - Release

Team Services / MyDriving

James Montemagno

HOME CODE WORK BUILD TEST RELEASE

Search work items

Explorer

Go here for help getting started with Release Management.

Search release definitions...

- Release Definitions
  - All release definitions
  - MyDriving.Services
  - MyDriving.Xamarin.Android
  - MyDriving.Xamarin.iOS
  - MyDriving.Xamarin.UWP

Definition: MyDriving.Xamarin.Android | Releases

Environments Artifacts Configuration Triggers General History

Save | Release

Add environment

Alpha Testers
1 / 1 tasks enabled
0

Beta Testers
0 / 1 tasks enabled
0

Push to Store
0 tasks
0

Add tasks

HockeyApp
Deploy MyTrips to HockeyApp

Deploy MyTrips to HockeyApp

HockeyApp Connection	Build HockeyApp (Manage)
App ID	$(HockeyAppId)
Binary File Path	$(System.DefaultWorkingDirectory)\MyDriving.Xamarin.Android/drop/bin/release/com.microsoft.mydriving.apk
Symbols File Path	
Native Library File Path	
Release Notes (File)	
Release Notes	Building IoT or Mobile solutions are fun and exciting. This year for Build, we wanted to show the amazing scenarios that can come together when these two are combined. So, we went and developed a sample
Publish?	☑
Mandatory?	☐
Notify Users?	☐

Download Restrictions

Tag(s)	
Team(s)	
User(s)	

Control Options

Enabled	☑
Continue on error	☐
Always run	☐

Replace with markdown to show in help

Distribute & Monitor

Distribute

Monitor

HockeyApp

Improve your apps with Real-time monitoring.

Native & Managed Crash Reports from C# API

HockeyApp

MyDriving iOS | Beta

Version 1.0

Crash Group | Overview | Crash Logs 134 | Bug Tracker

Add Annotation | Status: open

- [DelegatePage ThrowIndexOutOfBoundsException(System.Object se…

System.IndexOutOfRangeException: Index was outside the bounds of the array.

`DelegatePage.cs, line 105`

First Occurred	Last Occurred	Count	Jailbroken
22 Apr 2016, 20:20	24 Apr 2016, 20:40	134	0%

Crashes per Day

89	134	134
Last 24 Hours	Last 7 Days	Last 30 Days

Impacted Users per Day

89	119	119
Last 24 Hours	Last 7 Days	Last 30 Days

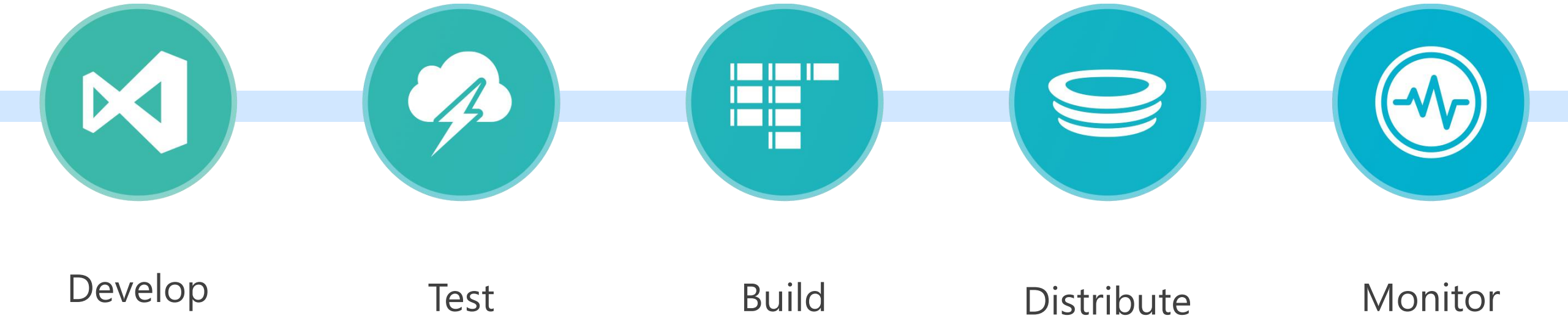
Top Affected OS's

9.3.1
Top Affected OS

Top Affected Devices

iPhone 6s
Top Affected Device

Xamarin – Your Complete Mobile Solution



Get Started Today
xamarin.com

Xamarin University

Unrivaled Mobile Development Training

Live unlimited mobile development training from mobile experts, in your time-zone, on your schedule, and as often as you'd like.

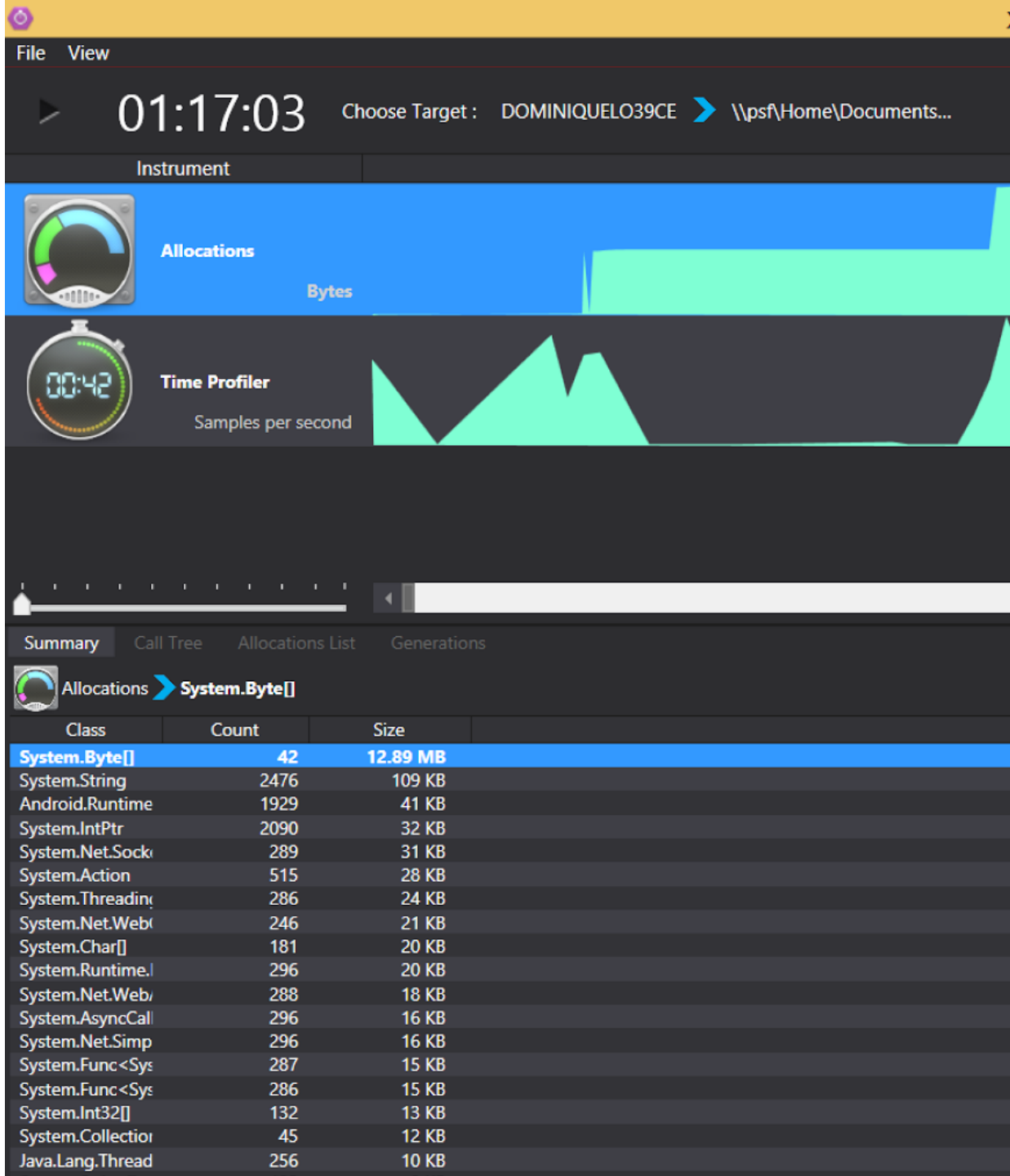
Free 30 Day Trial - xamarin.com/university

Additional Content

Xamarin Profiler Preview

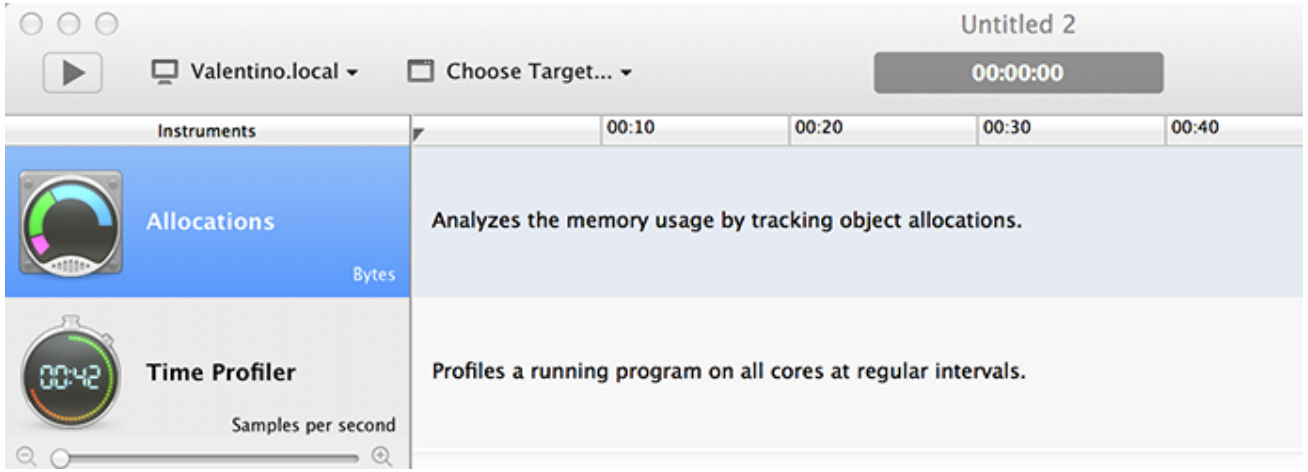
- Analyze & Polish Apps
- For all platforms!
  - iOS
  - Android
  - Xamarin.Forms
- Available on:
  - Mac or PC

Xamarin.com/profiler

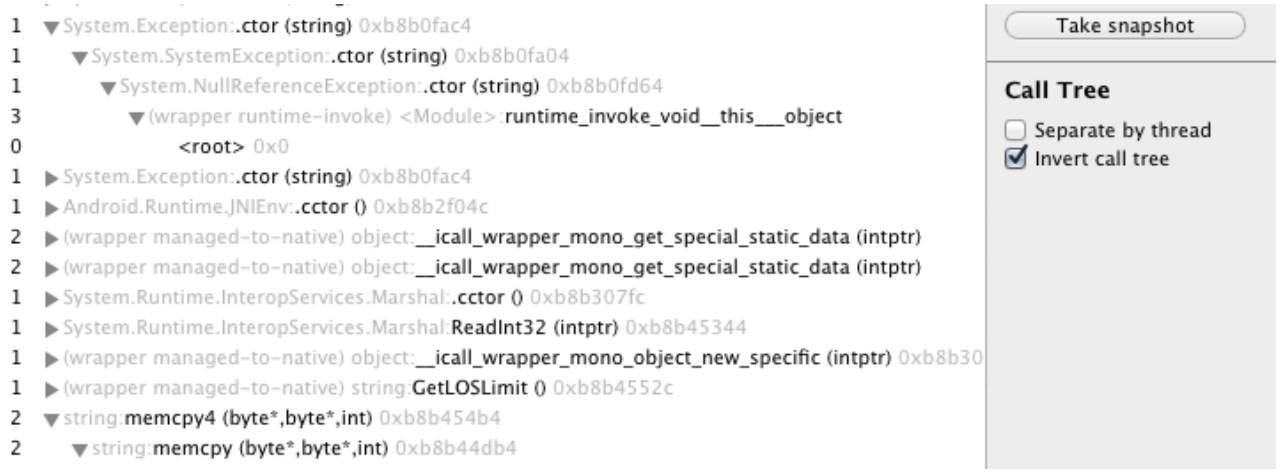


Results that matter

- Two key instruments



- Call Tree



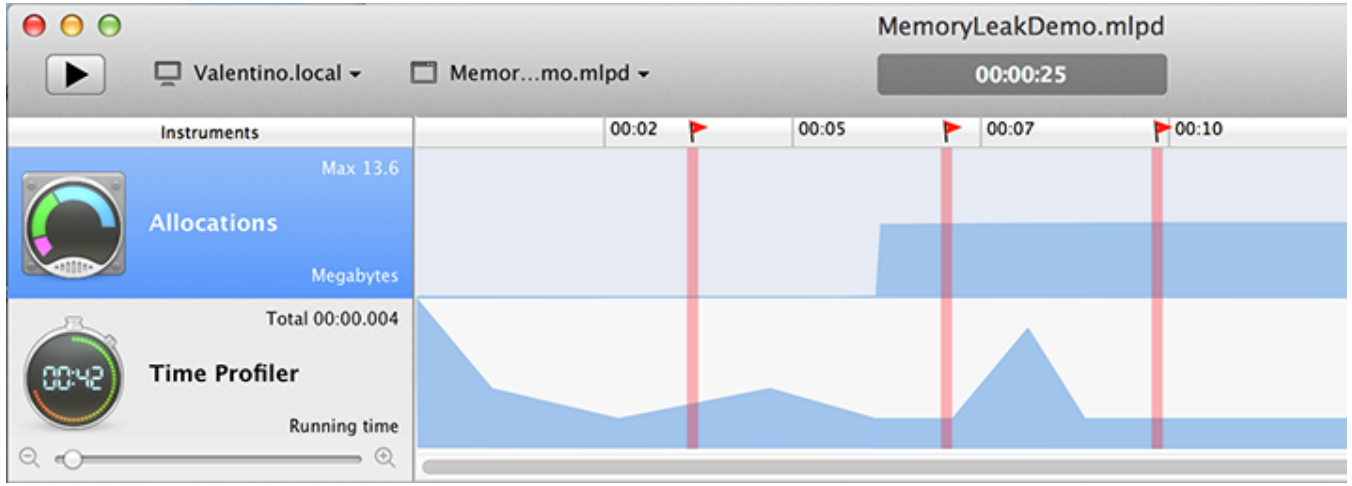
Results that matter

- Sampling

Time Profiler ▸ Sample List ▸ All samples Filter

# Hits	Running Time	Name
910 (35.88%)		Unknown
498 (19.64%)		emit_method
317 (12.50%)		-[GLKView initWithCoder:]
209 (8.24%)		method_enter
185 (7.29%)		method_leave
144 (5.68%)		__recvfrom
39 (1.54%)		mono_compile_create_var_for_vreg
34 (1.34%)		mono_method_needs_static_rgctx_invoke
29 (1.14%)		CFRunLoopTimerCreate
25 (0.99%)		gc_reference
21 (0.83%)		__workq_open
14 (0.55%)		mach_absolute_time
14 (0.55%)		__mkfifo_extended

- Snapshots

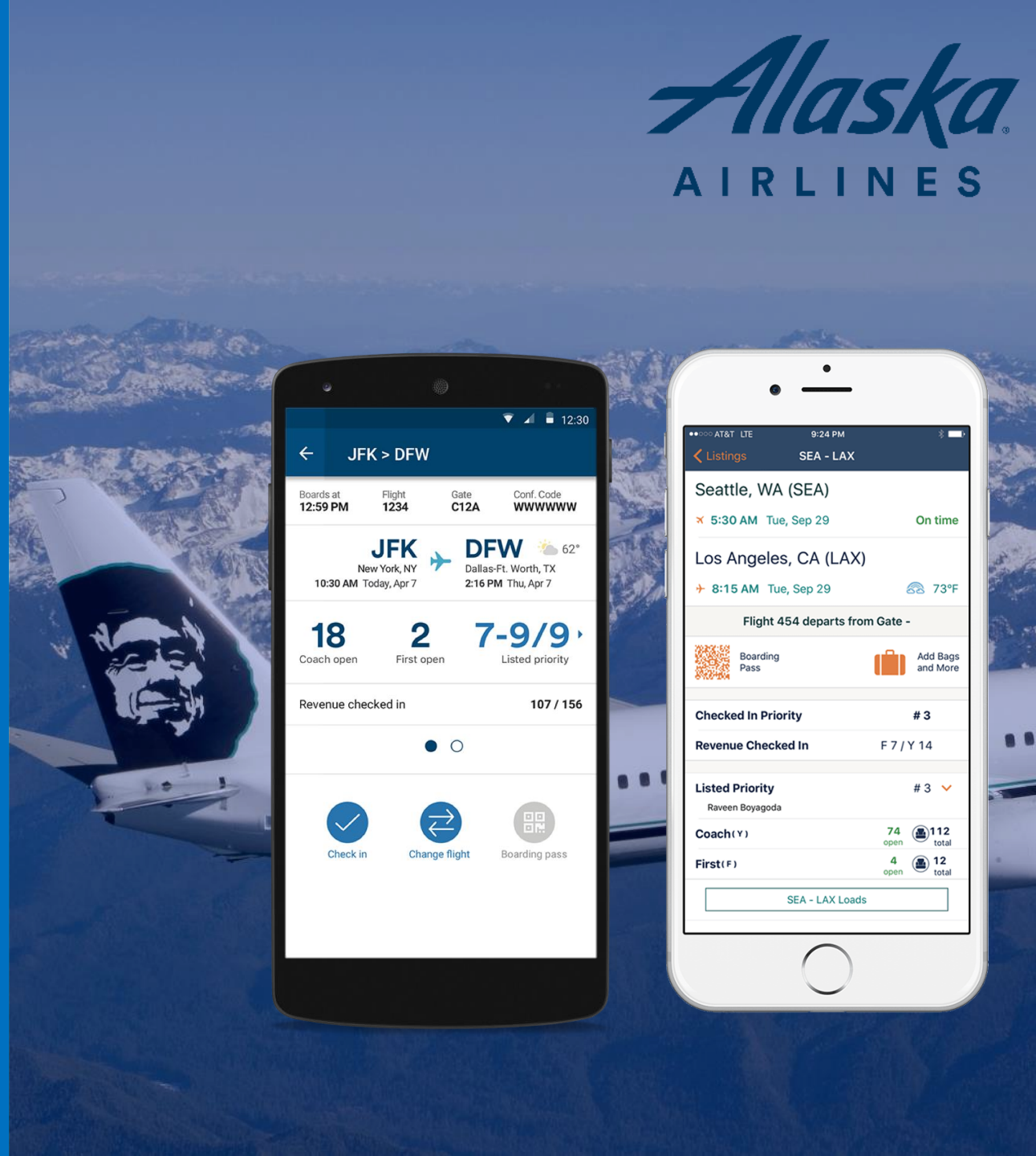


Alaska Airlines

Alaska Airlines is on a mission to be the easiest airline to fly by 2017. Its suite of mobile apps is helping make this a reality, allowing Alaska's 15,000 team members to access information from their device of choice, get company news, take advantage of their standby travel benefits, and provide in-app ratings to help the development team make improvements.

- Mirrors consumer UX, with mobile check-in and boarding passes
- Roadmap of 30+ employee apps (Android, iOS, and Windows)
- Employees are more engaged, productive, and customer focused
- Fully automated DevOps process with Visual Studio, Xamarin, Azure, Xamarin Test Cloud, Visual Studio Team Services, and HockeyApp

Available on iOS



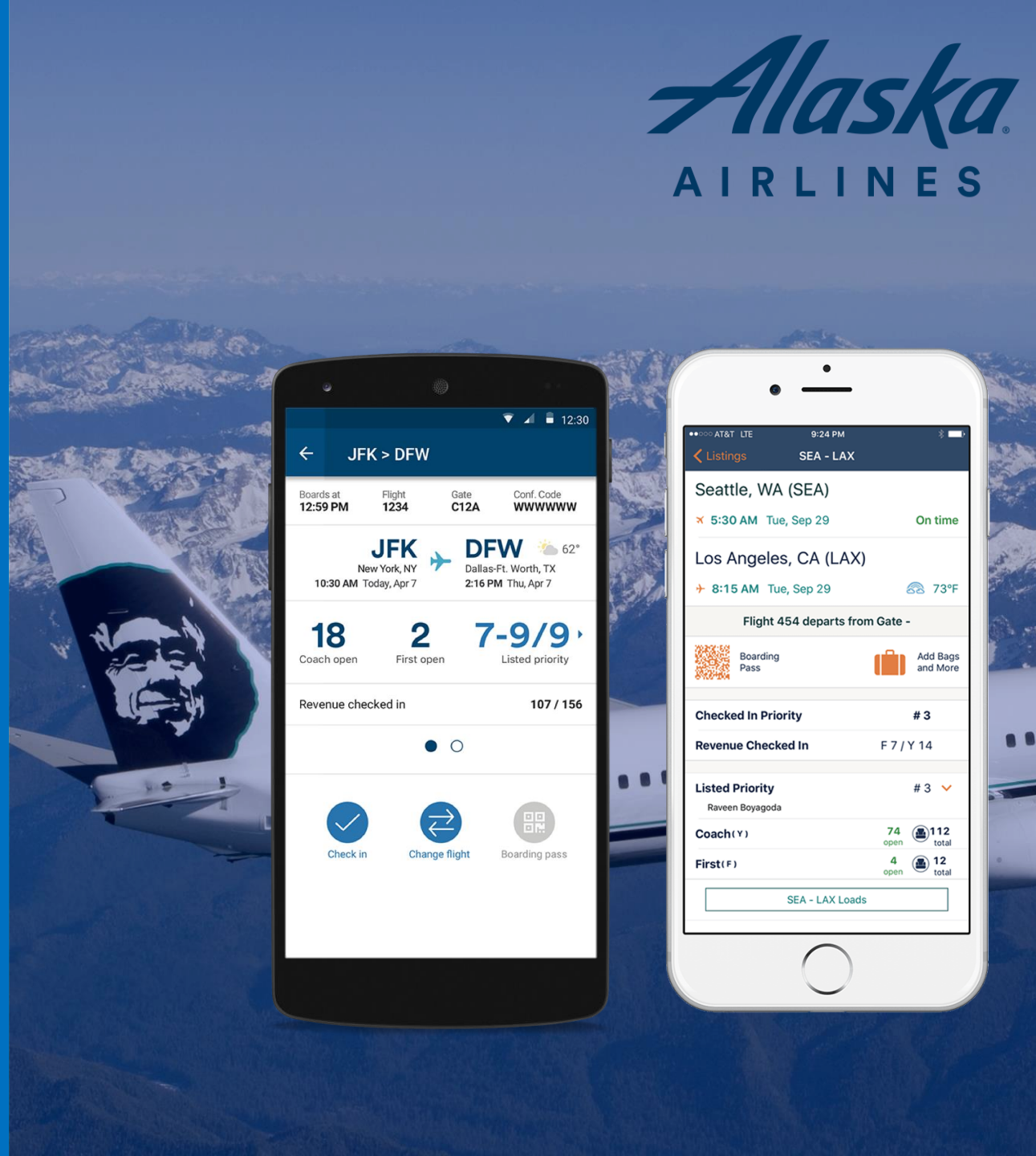
Alaska Airlines

"Mobile allows us to create a whole new experience for our employees and, with Xamarin and Microsoft, we're able to achieve our cloud-first, mobile-only vision.

Microsoft and Xamarin help us quickly deliver business value, getting high quality apps that perform in any scenario, on any device, to our thousands of staff members, while remaining confident that our internal data is safe and secure."

Mike Lorengo
Dir, ITS Architecture & Strategy
Alaska Airlines

Available on iOS



Cinemark

Cinemark is a leading, state-of-the-art theater chain, with 500+ locations and $2.8 billion in annual revenue. With its Xamarin apps, Cinemark delights movie-goers and builds customer loyalty, making it easy to locate nearby show times, securely purchase tickets, and eliminate wait times.

- Ensures high quality user experience across platforms with Xamarin Test Cloud – 6700+ five-star Google Play ratings
- Taps into mobile unique functionality, including social sharing, GPS, and mobile payments
- Introduced loyalty program to create direct relationship with previously anonymous movie goers, ensuring repeat visits
- Innovative "Cinemode" silences phones for movie duration, gamifying experience and rewarding customers with ticket coupons

Available on iOS

Cinemark

"We tried using HTML5 solutions, but it was challenging to debug the apps and make them work properly on each platform.

We've had a lot of feedback about our Xamarin app being more responsive, and people like the native interfaces."

Joe Dan Galyean
VP Engineering
Cinemark

Available on iOS

Thank You!

Joshua Joshua
Tech Evangelist

Joshua.Joshua@studentpartner.com www.joshuajosh.com.ng @JoshuaJsmart