Continuous Delivery for iOS, Android & Windows Apps using Xamarin, VSTS & HockeyApp

Nish Anil
Senior PM – Xamarin @ Microsoft
@nishanil

Microsoft – Your Complete Mobile Solution

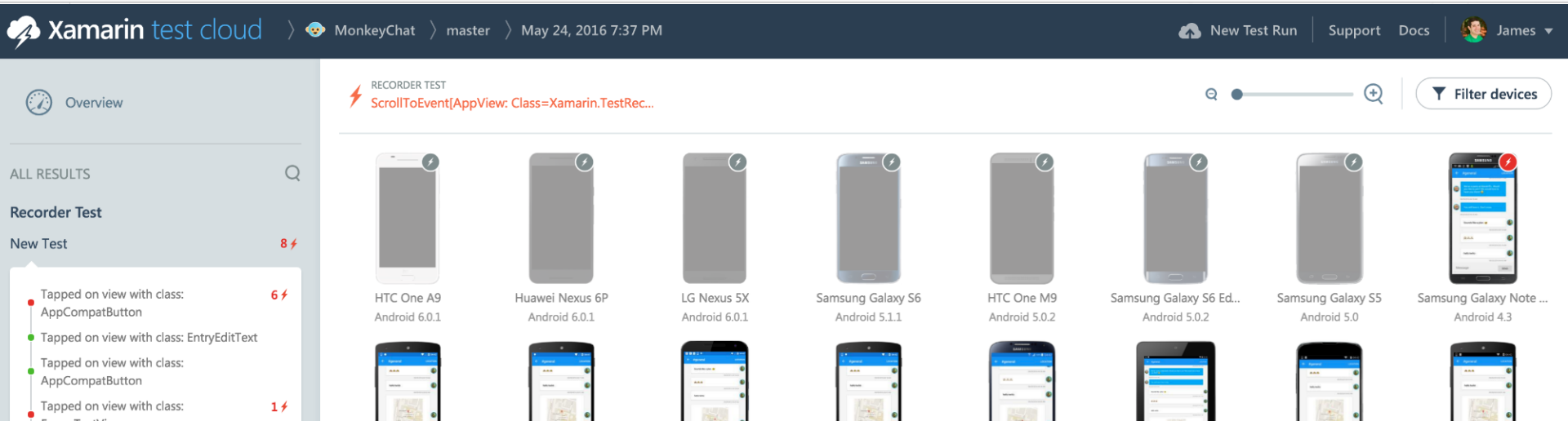
Develop

Test

Build

Distribute

Monitor



Test

Challenges

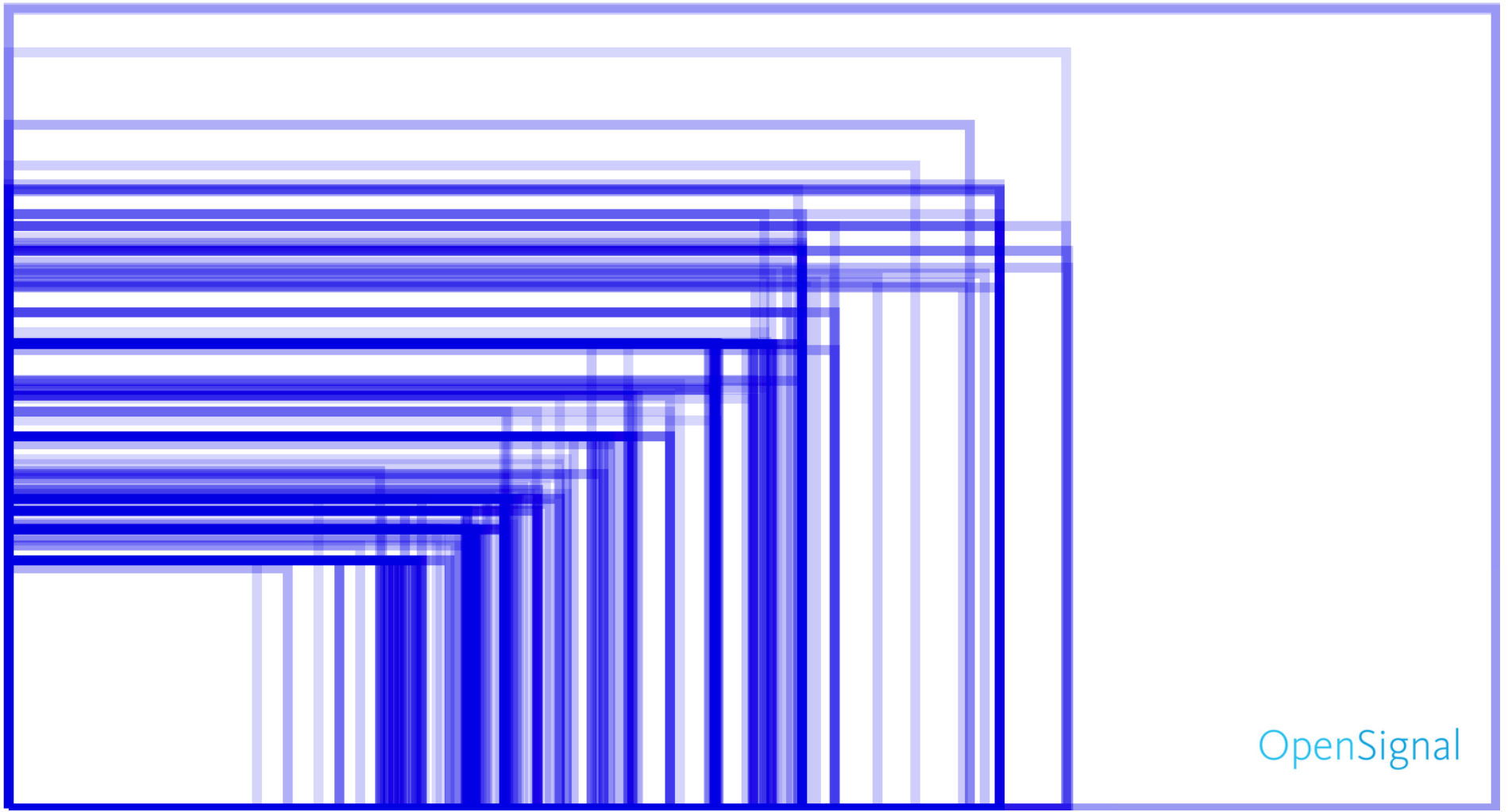
- Device Fragmentation
- App Complexity
- Fast Release Cycles
- Short Sessions
- High Mobile User Expectations

On Mobile, Quality is Hard

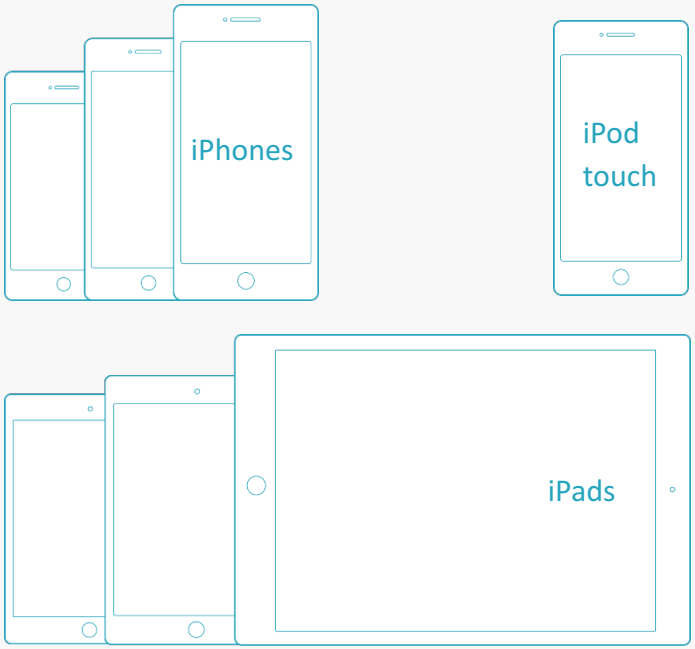
- 5 OS versions
- 20 Devices
- 20 Languages
- 35 Locales
- 6 Screen sizes

- 9 OS versions
- 24K+ Distinct devices
- 39 Languages
- 57 Locales
- 27 Screen sizes
- 1,294 Brands
- 6 Screen configurations

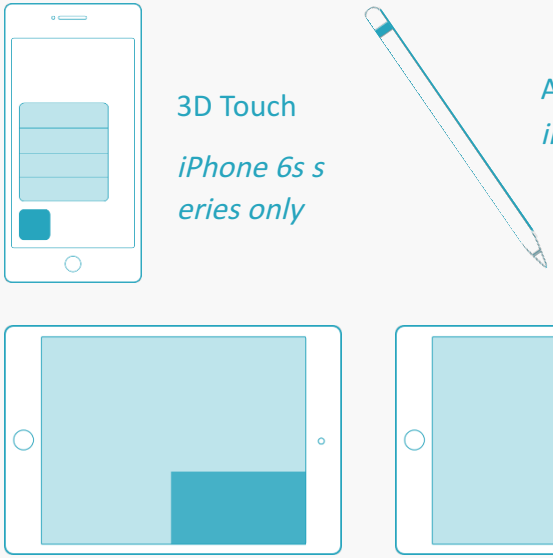

Galaxy S III
Galaxy S5
SM-G530H
Galaxy Note 3
Galaxy S4
Galaxy S4
Moto G
Moto G
GT-S7580
Moto G
D6503
SM-N9005
SM-G355H
Galaxy S II
SM-G7102
GT-I8262
GT-N7100
GT-S7582
G3
Galaxy S5
One
Galaxy Y
C6903
G2
K012
One (M8)
GT-I8552B
S6 Edge
A536
X
ZenFone 5
One
A369i
D2303
Z970
XL
GT-I9190
C5303
XT1080
XT1254
SM-T211
D5803
Moto E


OpenSignal

iOS Fragmentation: It does exist!



Form factors



iOS 9 features

Tap

Scroll

Swipe

Pinch

Multi Finger

Text Entry

Rotation

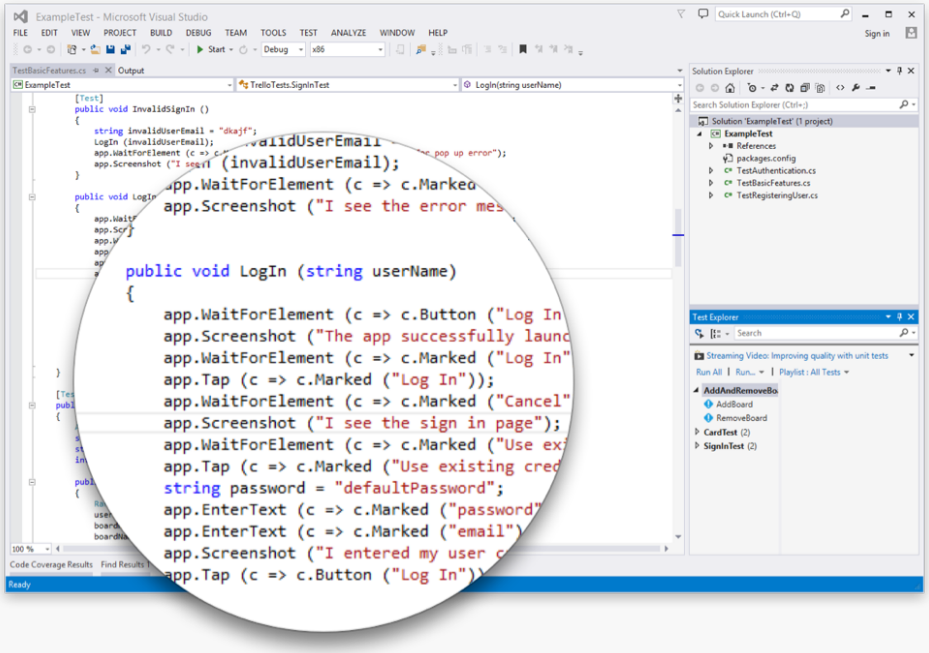
GPS

Language

Currency

Network Condition

Introducing Xamarin.UITest



Create Automated User Interface tests all in C#

Run directly from Visual Studio or Xamarin Studio

FREE to use on a Simulator or Device

Works on ANY app:

- Native
- Hybrid
- Xamarin

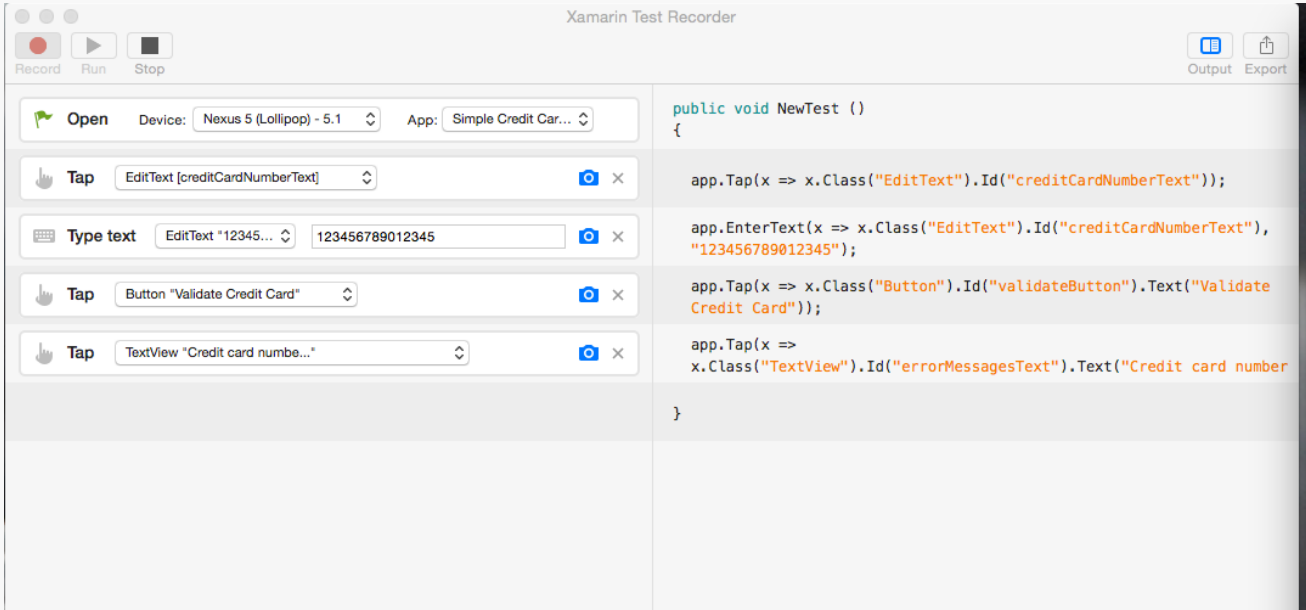
Every App, Every Language

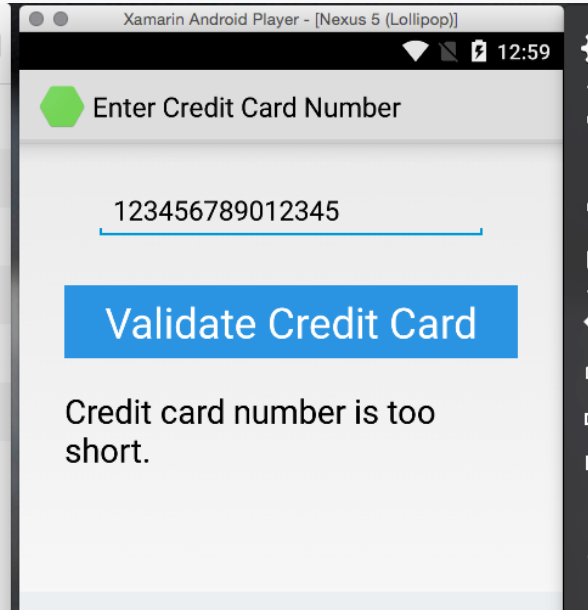
- Native Objective-C, Swift, Java
- Native C# with Xamarin
- Hybrid: PhoneGap, Cordova, etc
- Anything Else!

Let’s build our first Xamarin.UI Test!

Test Recorder for Mac

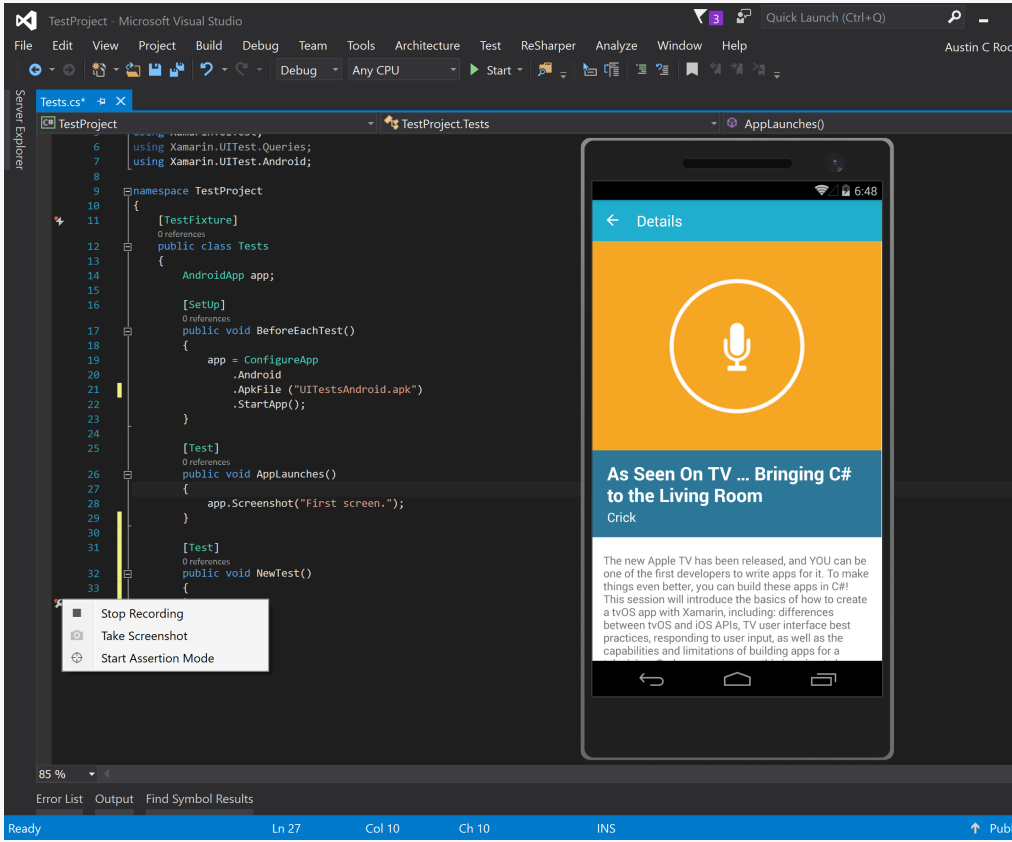
- Record UITests for iOS and Android apps
- Replay or ship to Test Cloud





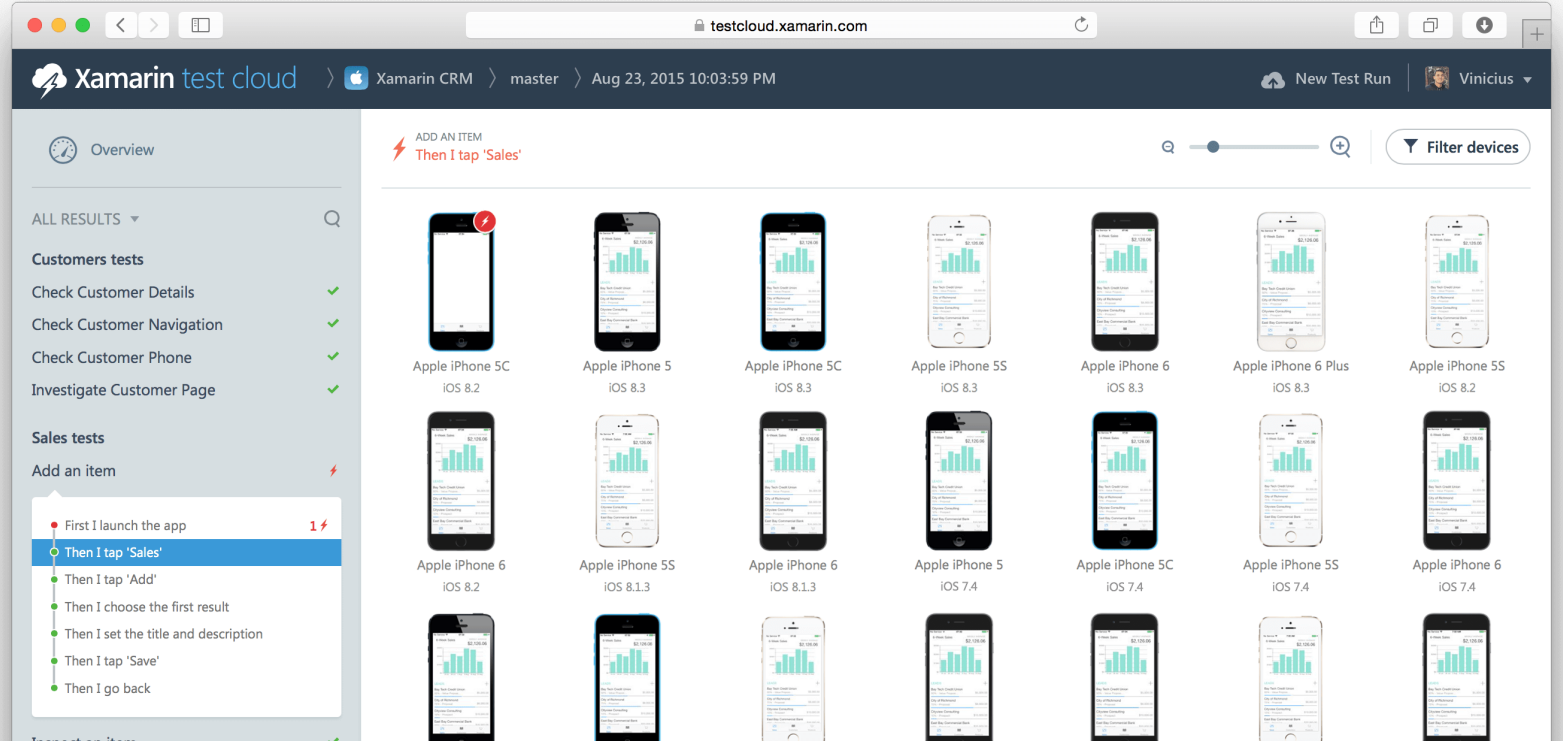
Test Recorder for Visual Studio

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

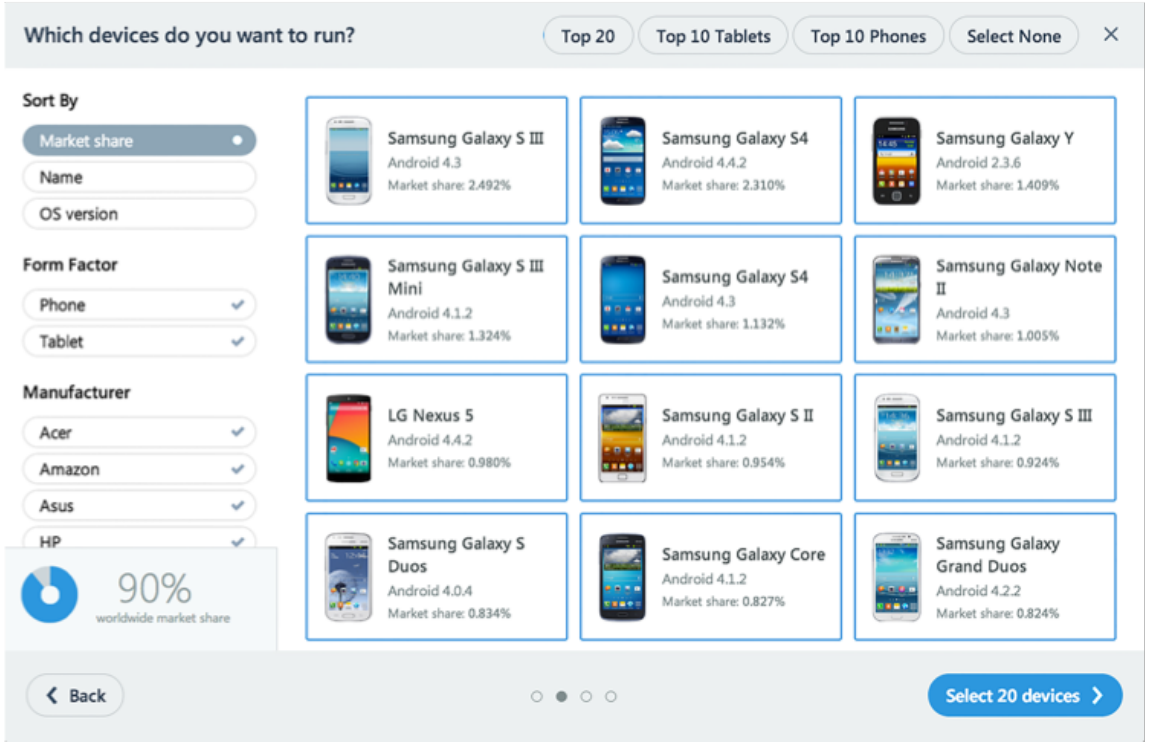


Xamarin Test Cloud

xamarin.com/testcloud



Devices on Demand



Multiple Supported Frameworks

Xamarin.UITest

- C#
- Run with Nunit
- XS & VS
- SpecFlow support

Calabash

- Ruby
- BDD, run with cucumber
- RubyMine or CLI

Appium

- Selenium or Webdriver
- Java, Python, JS

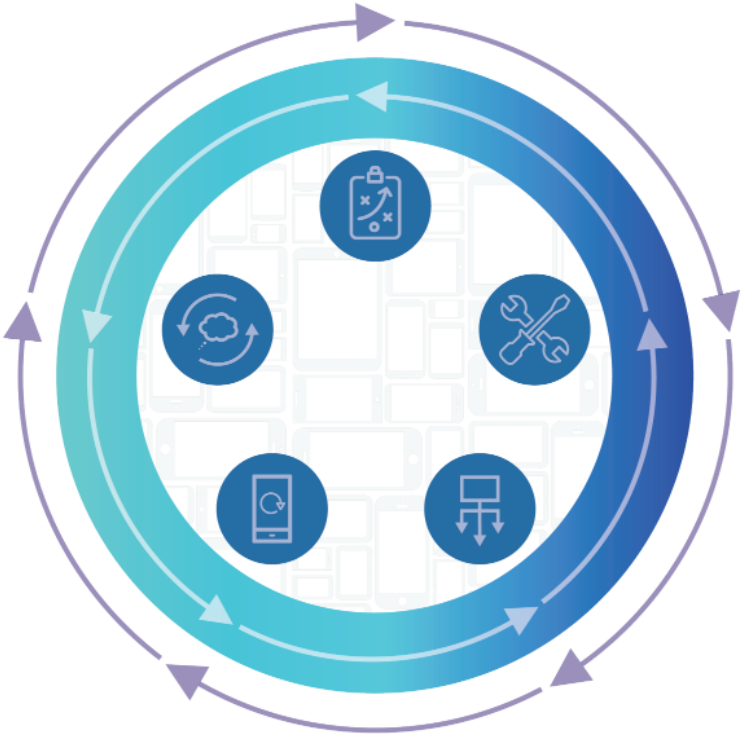
To The Test Cloud

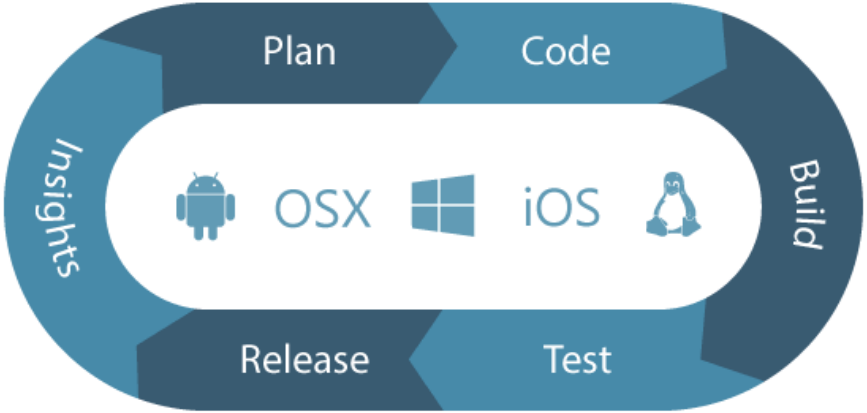
Build

Distribute

Monitor

What is DevOps ?




Plan
Code
Build
Test
Release
Insights
OSX
iOS

“DevOps is the union of people, process, and products to enable continuous delivery of value to our end users.”

- Donovan Brown

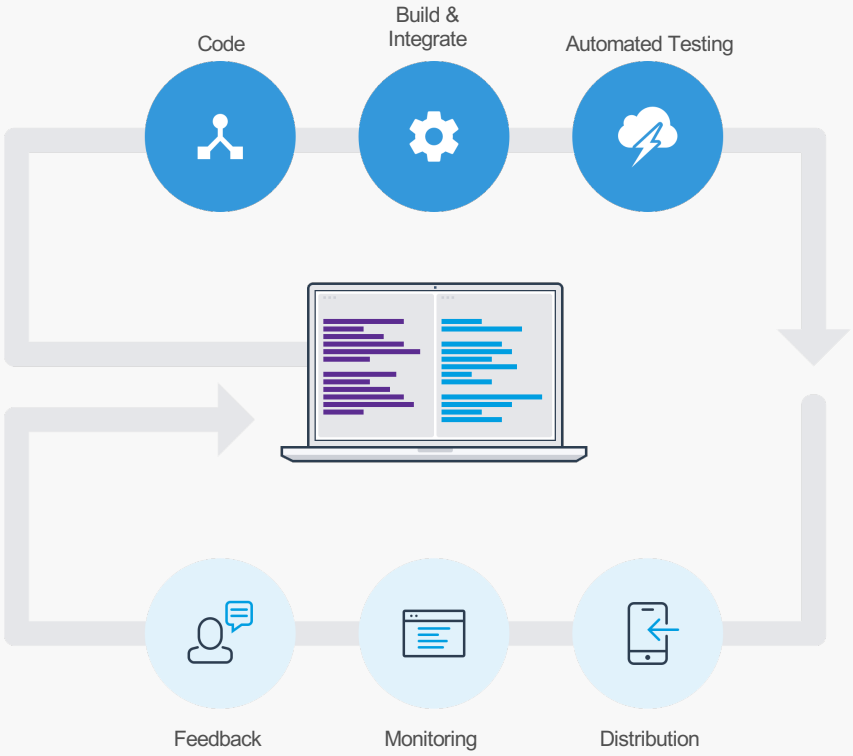
Senior DevOps Program Manager,

Microsoft

Mobile DevOps

“Mobile” DevOps

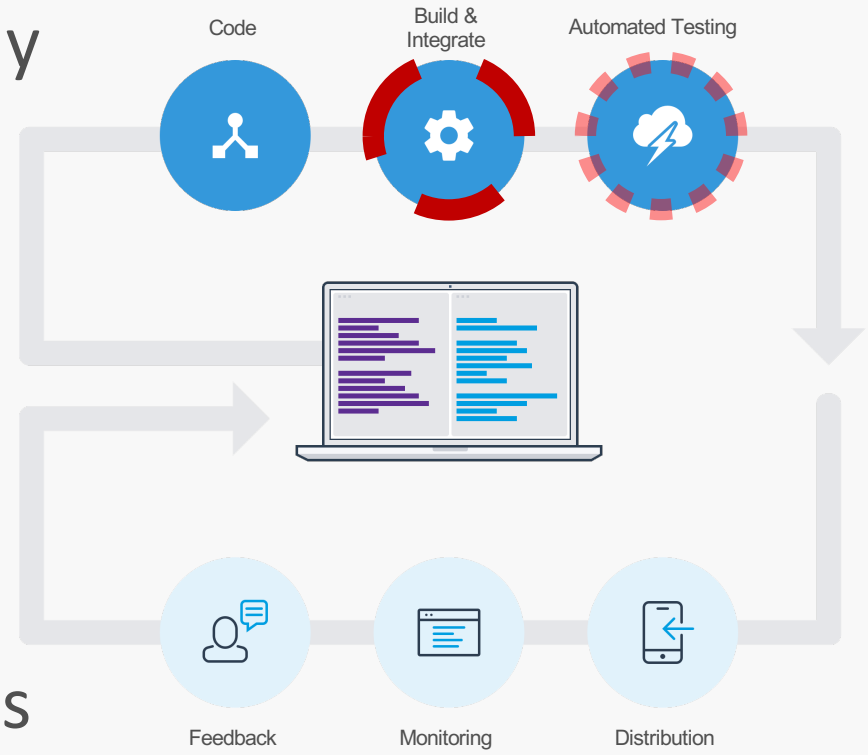
- Streamline
  - Frequently releasing value to users
  - Constantly maintaining quality
  - Monitoring app health & engagement in real-time
- Automating the process
  - Continuous Integration (CI)
  - Continuous Deployment (CD)
  - Continuous Monitoring (CM)



Continuous Integration

Continuous Integration

- Every Commit…Continuously
- Build
- Integrate
  - Version Bumping
  - Replacing Private Variables
  - Testing
    - Unit Tests
    - UI Tests
  - Other SDKs
- Sign, Package, Drop Artifacts



Visual Studio Team Services (VSTS) / TFS

Version Control

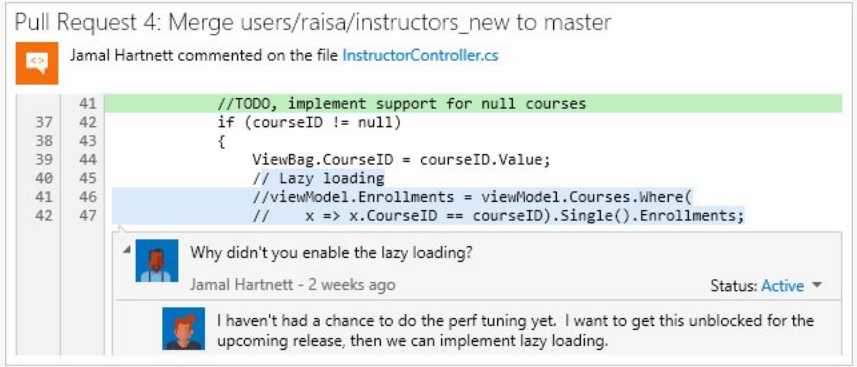
- Git and Team Foundation Version Control (TFVC)
- Unlimited private repos

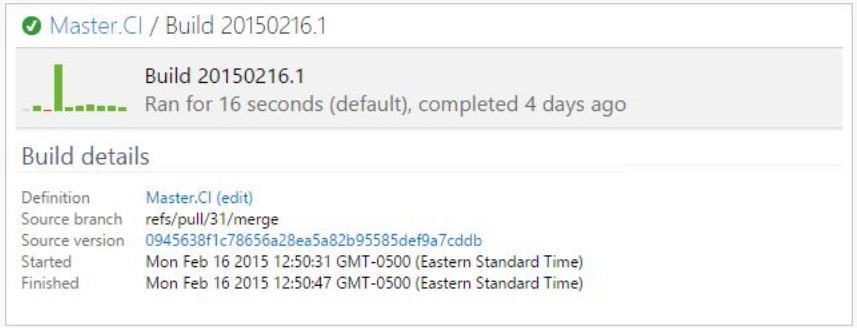
Agile Planning

- Work item tracking
- Kanban, backlogs, task boards

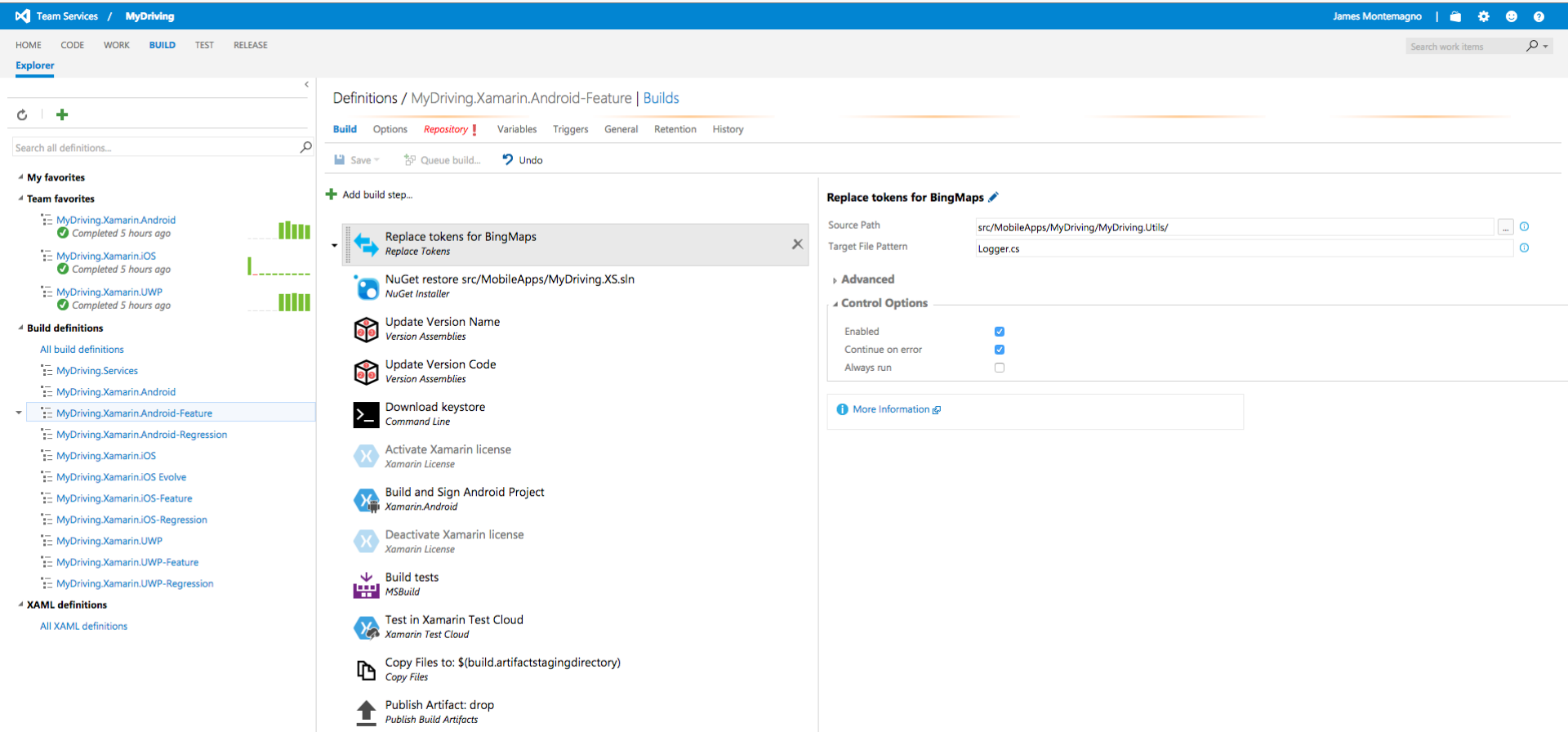
Continuous Integration

- Cloud-hosted build agents
- Build support for all mobile platforms
- Integration with test clouds





Visual Studio Team Services - Build

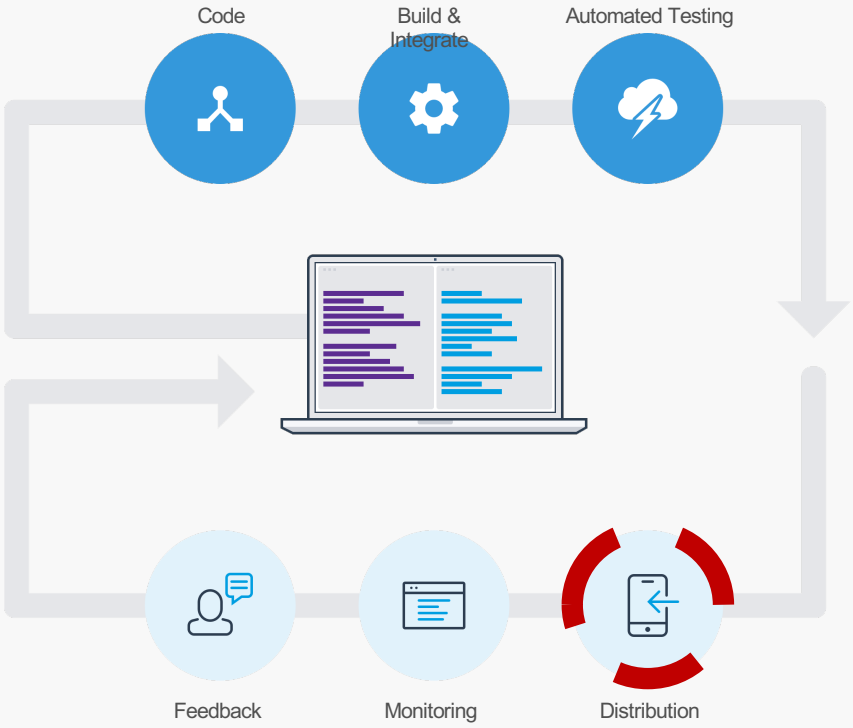


Demo – VSTS end to end

Continuous Deployment

Continuous Deployment – Alpha/Beta

- Development Builds
- Send to Testers
- Release to App Store
- Promote App Store Channel
  - Alpha -> Beta -> Production



HOCKEYAPP

HockeyApp

Distribution

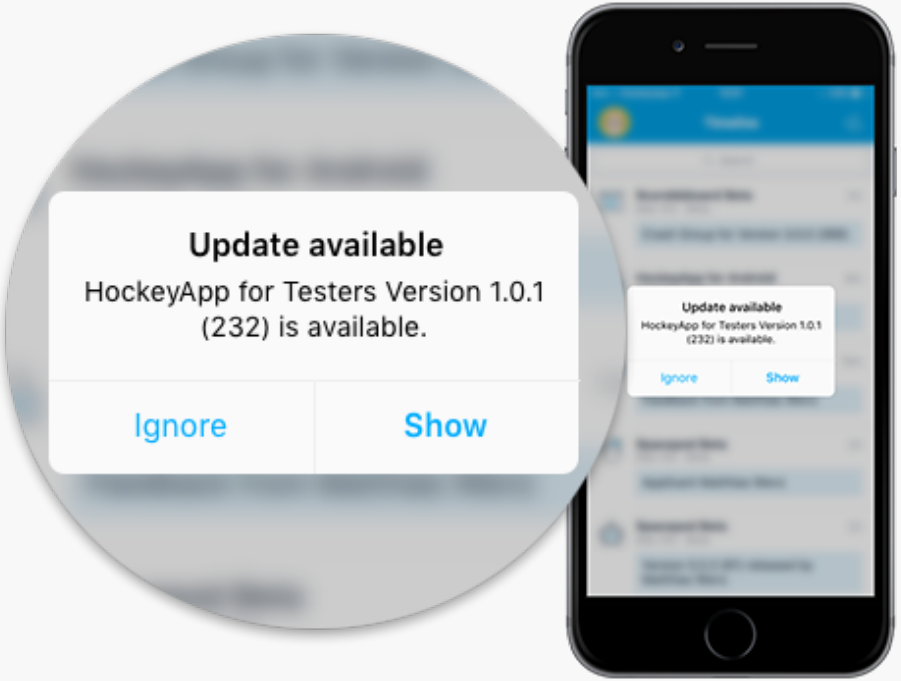
- In-house app store
- Notify testers of updates

Crash Reporting

- Real-time collection
- “Symbolicated” stacks

Feedback

- Gather in-app feedback
- Respond directly to testers

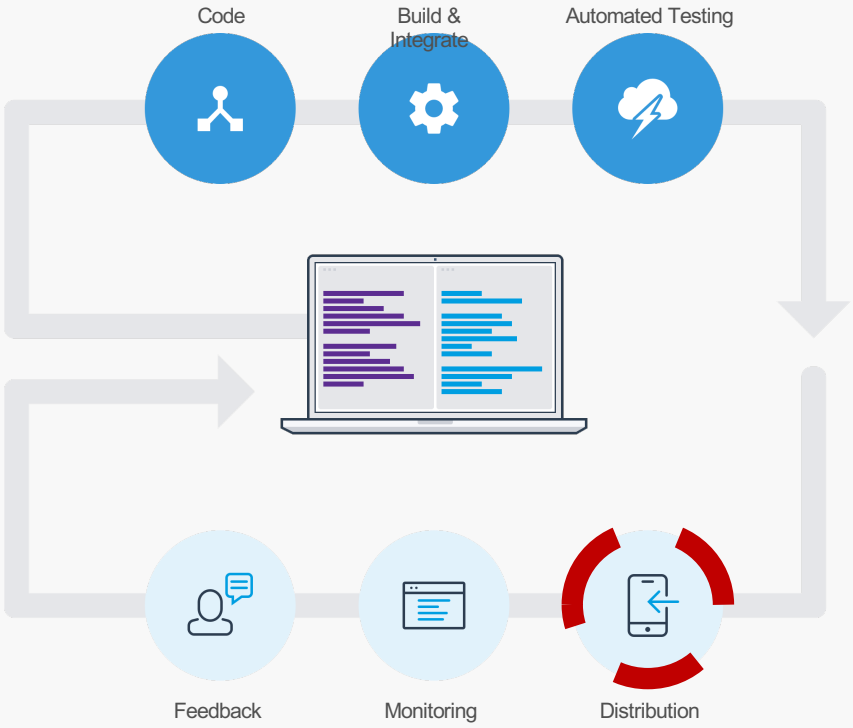


Continuous Deployment - Production

How can you automate your production deployments once a release is signed-off?

Recommendations

1. Setup a multi-environment release process (e.g. beta and production)
2. Assign the necessary stakeholders to sign-off on beta/staging builds
3. Automate as much of this workflow as possible



VSTS – Release Management

Continuous Delivery

Define per-environment release pipeline

Trigger manually or via successful builds

Approval Policies

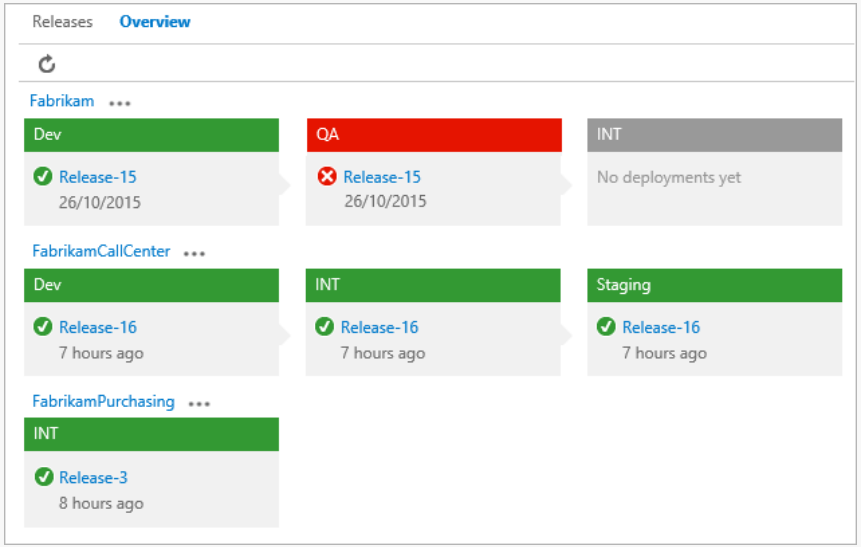
Configure per-environment sign-offs

Manual validation requirements

Release Visibility

View per-environment release status

Track release back to commits



Demo – HockeyApp

Get Started Today

testcloud.xamarin.com
visualstudio.com
hockeyapp.net

More!

Visual Studio	aka.ms/vsandxamarin
Xamarin	aka.ms/gettingstartedwithxamarin
GitHub	github.com/xamarin
twitter	twitter.com/msdevindia
facebook	facebook.com/MicrosoftDeveloper.India/
Microsoft	Booth # 10

Thank you. Questions?

Nish Anil
Senior PM– Xamarin @ Microsoft

nish@microsoft.com

@nishanil