

Nathan Westfall  
Software Engineer  
@timeawastin  
[github.com/nwestfall](https://github.com/nwestfall)  
[theengineerofwhat.com](https://theengineerofwhat.com)

---

# INTRODUCTION TO MOBILE UI AUTOMATION TESTING WITH APPIUM

# 4 TYPES OF TESTING

- ▶ Unit Testing
  - ▶ Breaks code into smaller groups
  - ▶ Fast to run and can be used for constant regressions
- ▶ Performance Testing
  - ▶ Mobile Specific -> Performance between devices
- ▶ UI Testing
  - ▶ This is the type that Appium falls in
- ▶ API Testing

# DIFFERENCE BETWEEN UNIT TESTING AND UI TESTING

- ▶ Unit Testing
  - ▶ Used for testing sections of code
  - ▶ Helpful for developers
  - ▶ Can be used along with code check-ins for quick regression
- ▶ UI Testing
  - ▶ Helps test a vast variety of factors (Screen size, hardware)
  - ▶ Lets you scale UI tests easily

# INTRODUCTION TO MOBILE UI AUTOMATION TESTING WITH APPIUM

adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)

## login sequence

✓ should swipe through product presentation (120022ms)  
✓ should skip product presentation (14231ms)  
✓ should reject wrong credentials (124189ms)  
1) should login into the app

adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
✓ should add a new dream (130047ms)  
✓ should change dream date (115472ms)  
✓ should add a reason to a dream (116847ms)  
✓ should add steps to a dream (117117ms)  
✓ should remove a dream (137117ms)

## log out user

adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)  
adding and manipulating a dream  
✓ step should result in an alert when there is no dream (10003ms)



## Automation for Apps

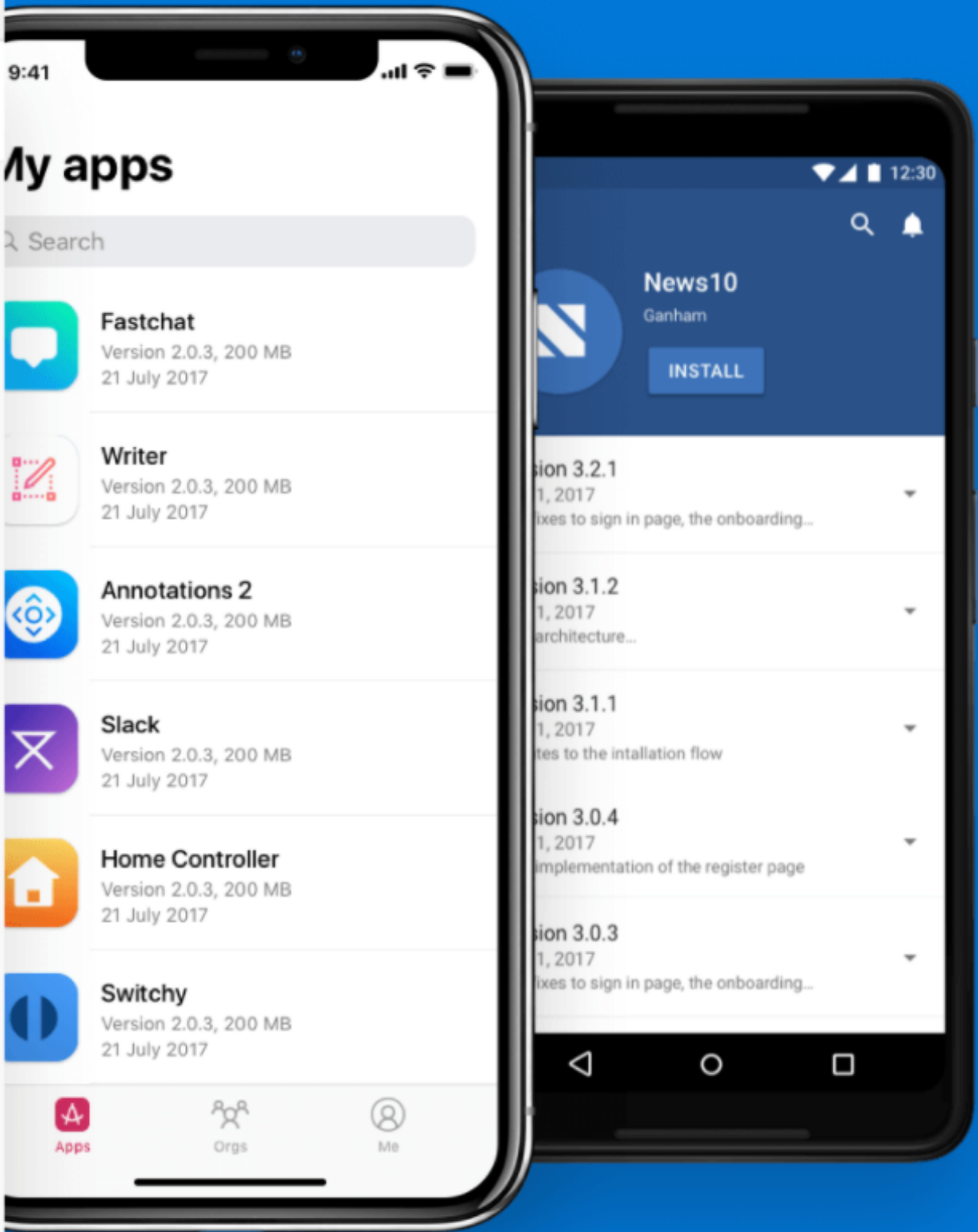
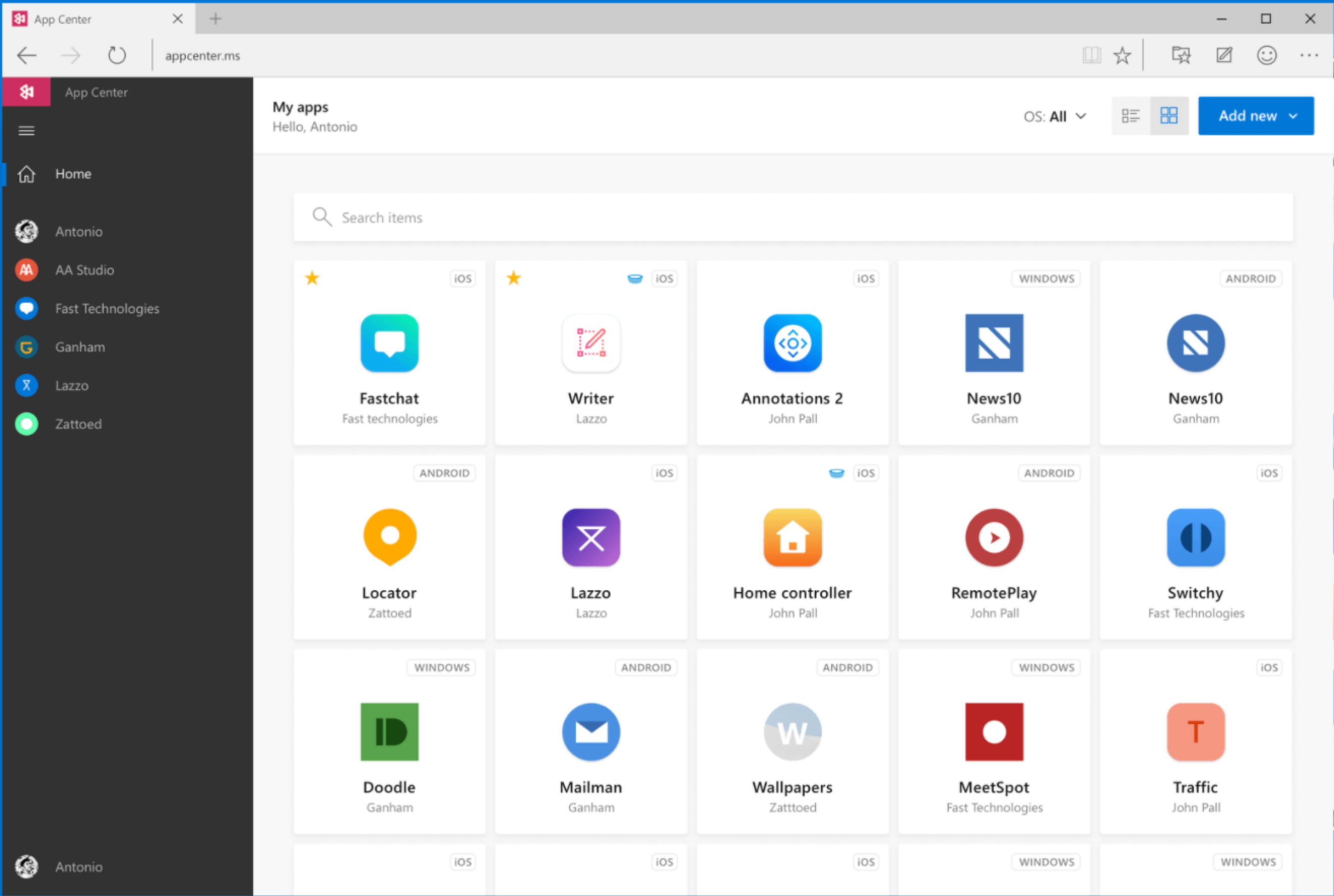
Appium is an open source test automation framework for use with native, hybrid and mobile web apps. It drives iOS, Android, and Windows apps using the WebDriver protocol.



# INTRODUCTION TO MOBILE UI AUTOMATION TESTING WITH APPIUM



## Visual Studio App Center



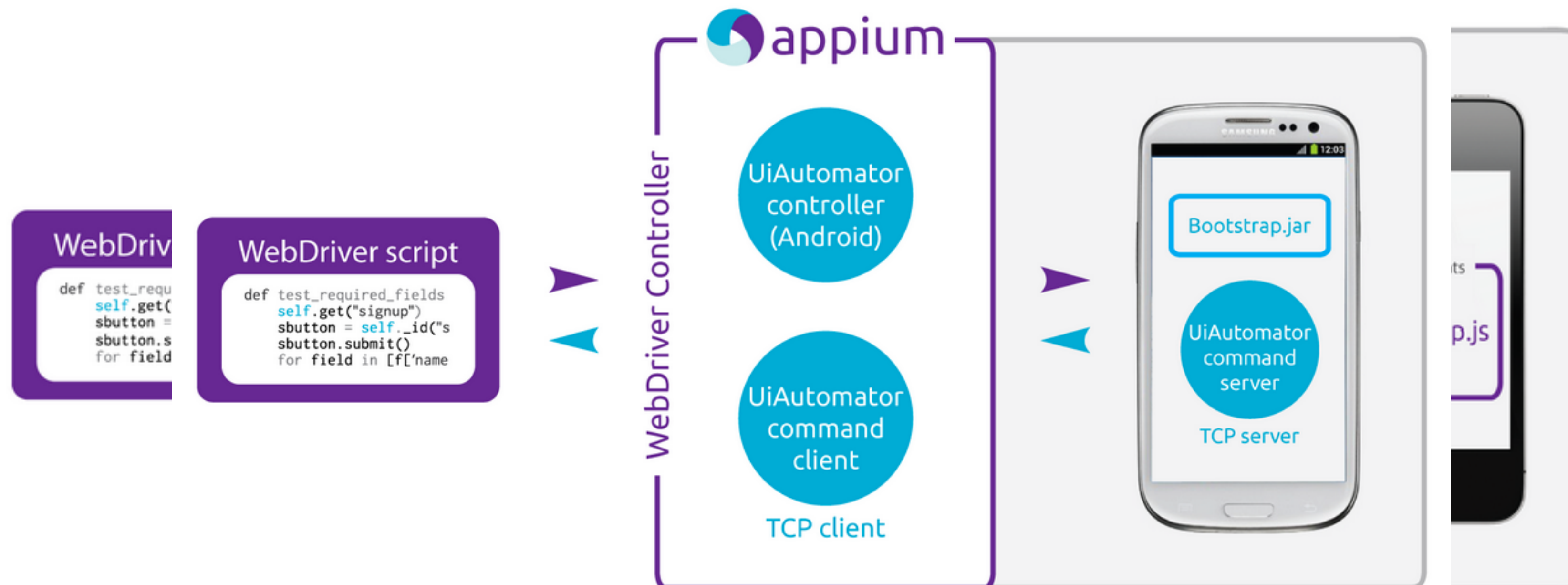
# AWS Device Farm

Improve the quality of your iOS, Android, and web applications by testing against real phones and tablets in the AWS Cloud.



## WHAT IS APPIUM?

- ▶ Appium is an open-source tool for automating native, mobile web, and hybrid applications
- ▶ Appium is "cross-platform": it allows you to write tests against multiple platforms using the same API



---

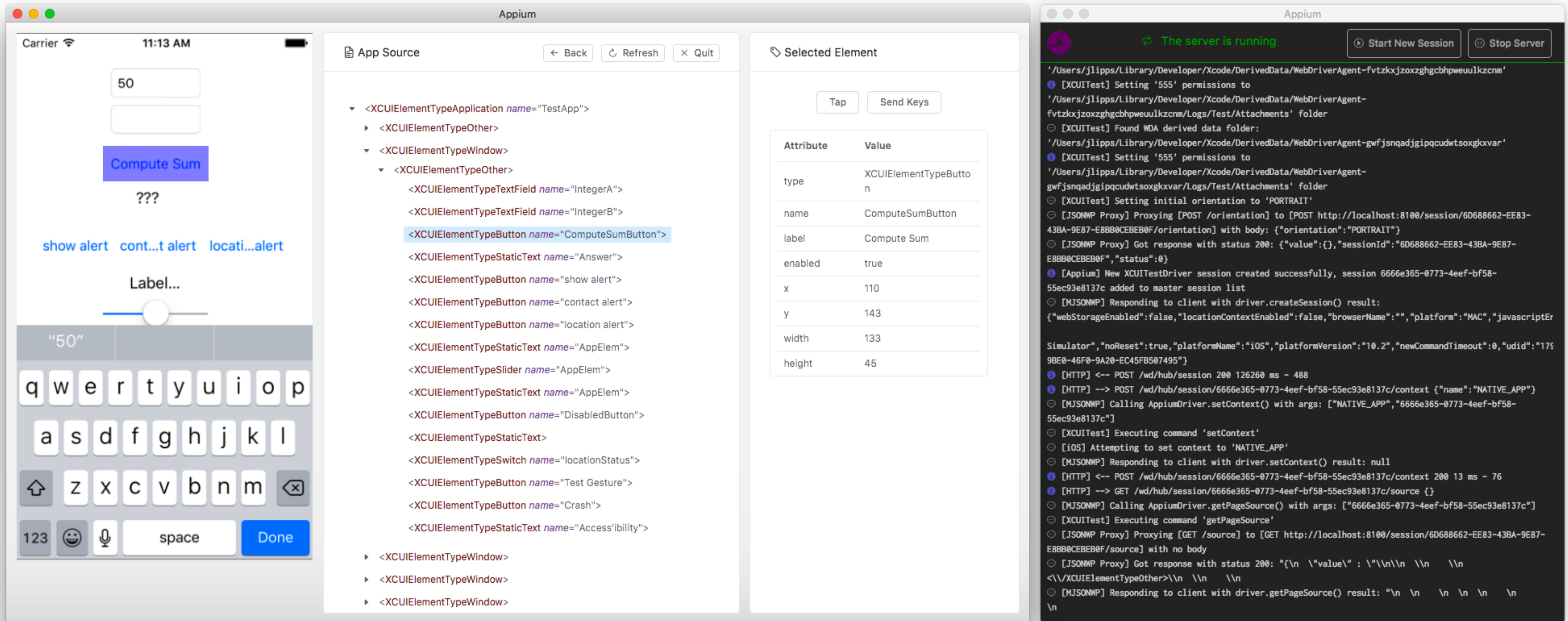
# DEMO #1



# INSTALLING APPIUM

- ▶ Appium Server
  - ▶ npm install -g appium
- ▶ Javascript Client
  - ▶ npm install webdriverio
- ▶ C# Client
  - ▶ Nuget Appium.WebDriver
- ▶ Python Client
  - ▶ pip install Appium-Python-Client
- ▶ Java Client
  - ▶ Maven + Gradle support

# APPIUM DESKTOP



---

# DEMO #2

# USING APPIUM WITH ANY LANGUAGE

- ▶ Any language that supports webdriver/http calls
  - ▶ C#
  - ▶ PHP
  - ▶ Java
  - ▶ Javascript
  - ▶ Python
  - ▶ Ruby
- ▶ Appium Desktop
  - ▶ Supports test recordings in...
    - ▶ JS w/ [webdriver.io](https://webdriver.io)
    - ▶ Java w/ JUnit
    - ▶ Python
    - ▶ Ruby



---

# DEMO #3

# AUTOMATING TESTS WITH APPIUM

- ▶ App Center
  - ▶ Currently only supports Maven and JUnit
- ▶ AWS Device Farm
  - ▶ Maven/JUnit and Python
- ▶ TFS/VSTS
  - ▶ Whatever language you want!
  - ▶ Allows you to test against in-house devices

---

# DEMO #4

Nathan Westfall  
Software Engineer  
@timeawastin  
[github.com/nwestfall](https://github.com/nwestfall)  
[theengineerofwhat.com](https://theengineerofwhat.com)

---

# QUESTIONS