TESTINGMIND

A single framework for Android, IOS and Web testing

#STS18

Manila
November 9th, 2018

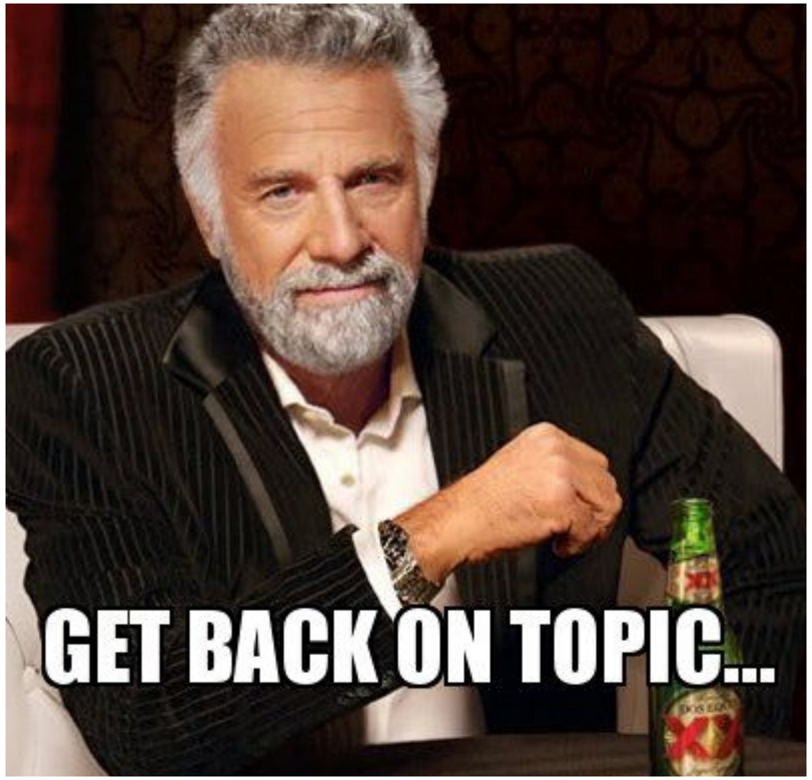
A short introduction

- Martin Schneider
- from Austria != Austr**al**ia → sorry, no kangaroos
- born in Salzburg aka the “The Sound of Music” city and Mozart’s birthplace

A short introduction contd.

- Master’s degree in Applied Mathematics
- various software development roles since ~ 2009
- focus on test automation since 2016
- currently living in Singapore and part of Carousell

UI testing multi-platform applications

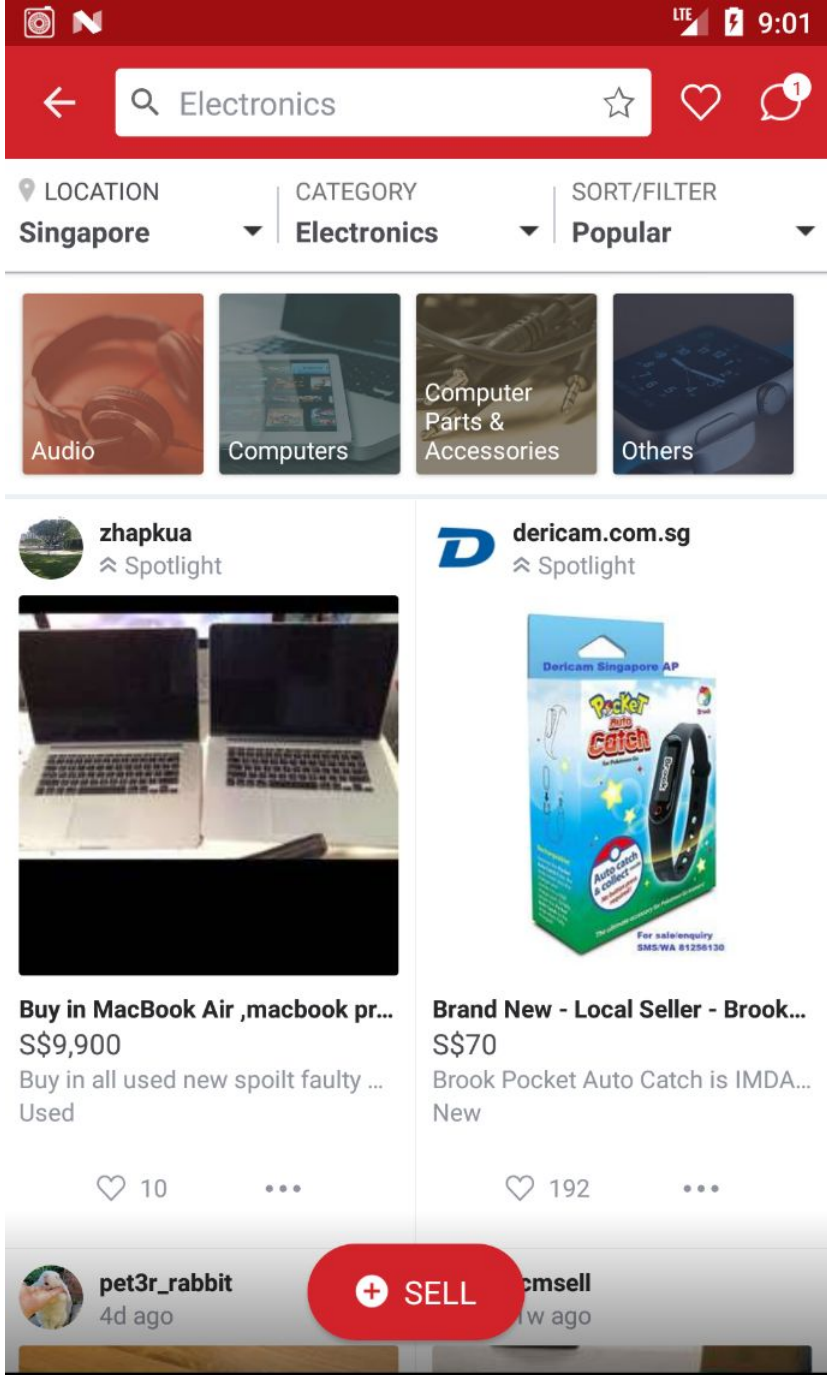


Multi-platform applications

- Multi-platform = Web + Mobile
- Mobile = iOS + Android

- Different “evolution” paths
  - Web application → mobile app (Amazon, eBay, Facebook, Wikipedia, PayPal, Twitter…)
  - Mobile app → web application (WhatsApp, Instagram, Carousell...)

For example:
  - eBay: Web 1995, iOS 2008, Android 2010
  - Carousell: iOS 2012, Android 2013, Web 2015


9:01

Electronics

LOCATION
Singapore

CATEGORY
Electronics

SORT/FILTER
Popular

zhapkua
Spotlight

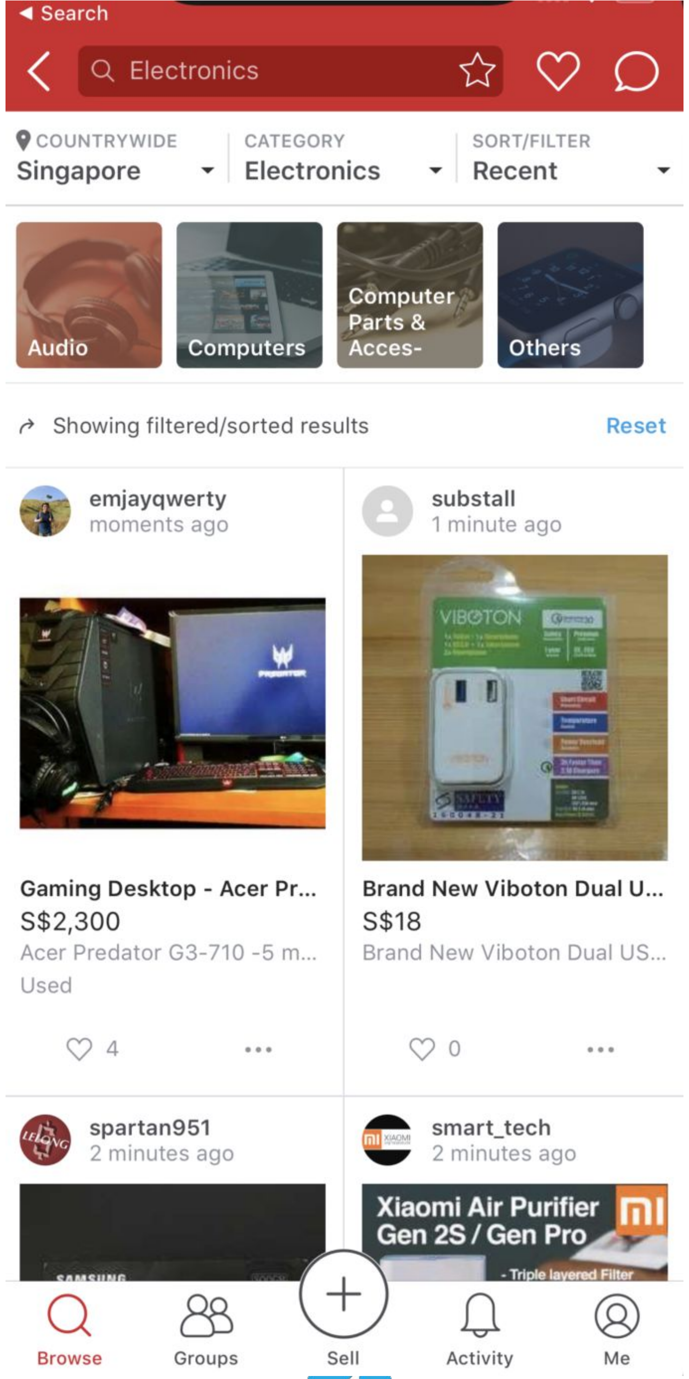
Buy in MacBook Air ,macbook pr...
S$9,900
Buy in all used new spoilt faulty ...
Used
10

dericam.com.sg
Spotlight

Brand New - Local Seller - Brook...
S$70
Brook Pocket Auto Catch is IMDA...
New
192

pet3r_rabbit
4d ago

SELL


Search

Electronics

COUNTRYWIDE
Singapore

CATEGORY
Electronics

SORT/FILTER
Recent

Showing filtered/sorted results
Reset

emjayqwerty
moments ago

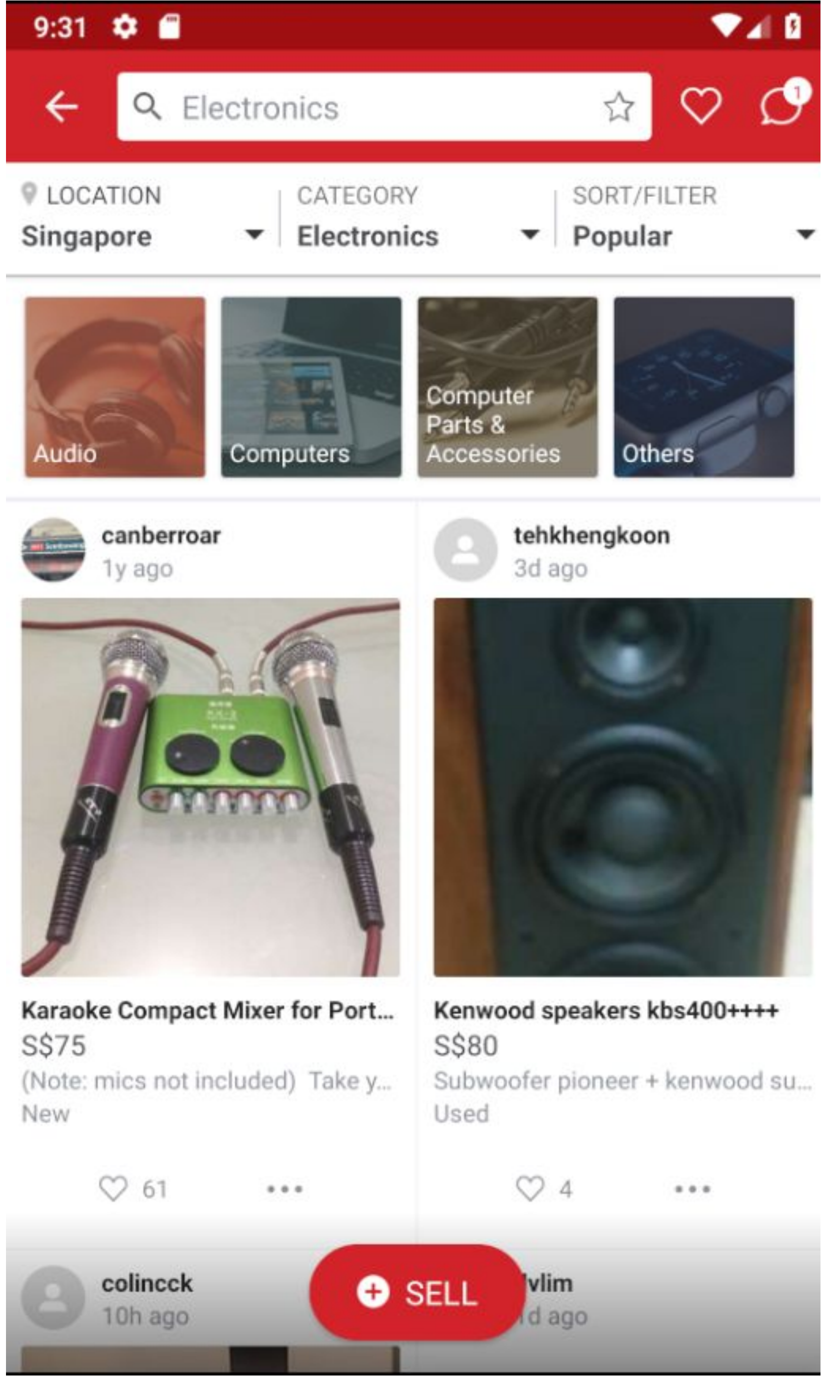
Gaming Desktop - Acer Pr...
S$2,300
Acer Predator G3-710 -5 m...
Used
4

substall
1 minute ago

Brand New Viboton Dual U...
S$18
Brand New Viboton Dual US...
0

spartan951
2 minutes ago

smart_tech
2 minutes ago


9:31

Electronics

LOCATION
Singapore

CATEGORY
Electronics

SORT/FILTER
Popular

canberroar
1y ago

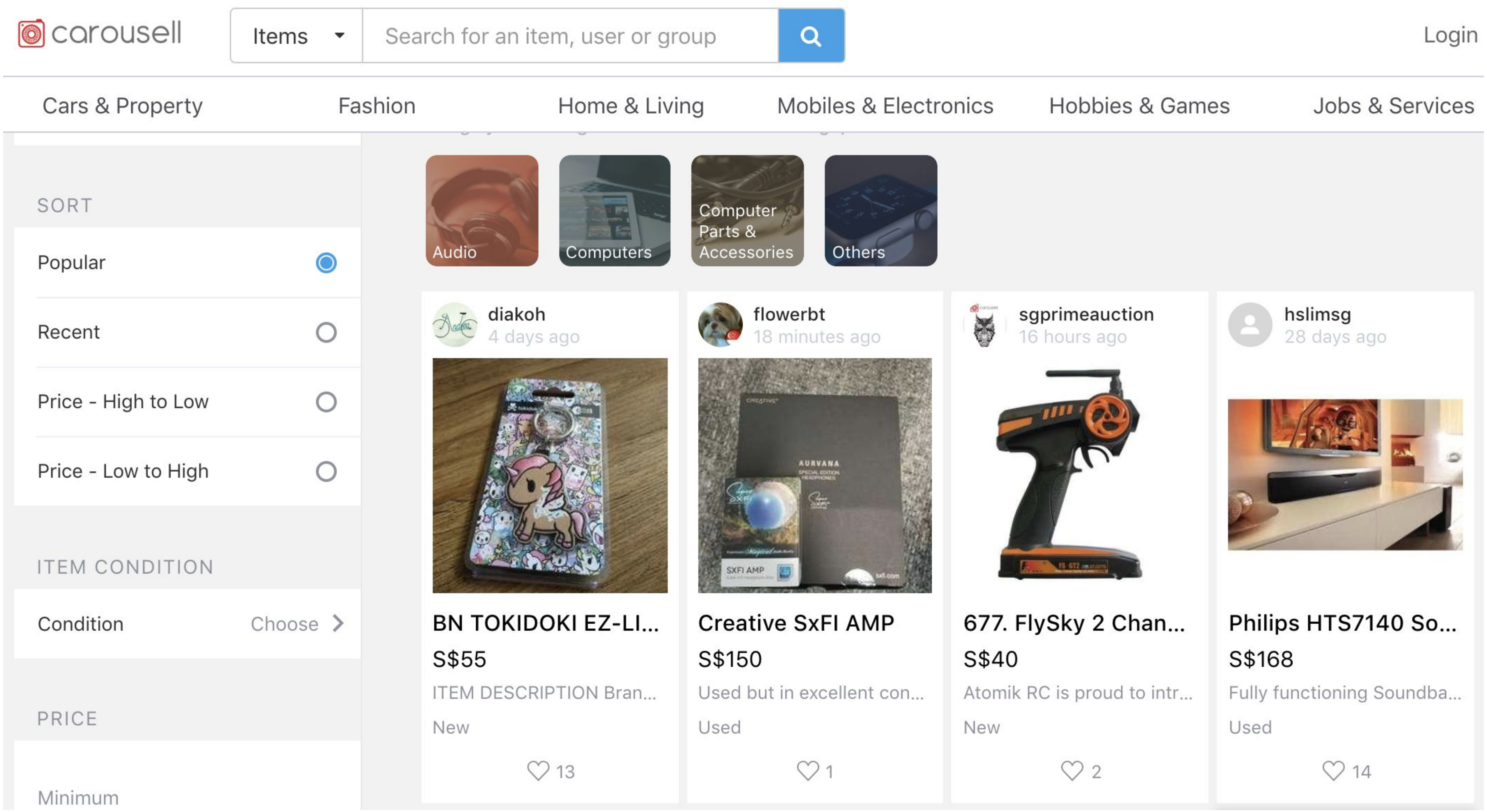
Karaoke Compact Mixer for Port...
S$75
(Note: mics not included) Take y...
New
61

tehkhengkoon
3d ago

Kenwood speakers kbs400++++
S$80
Subwoofer pioneer + kenwood su...
Used
4

colincck
10h ago

SELL



carousell

Items | Search for an item, user or group

Login

Cars & Property | Fashion | Home & Living | Mobiles & Electronics | Hobbies & Games | Jobs & Services

SORT

Popular ●

Recent ○

Price - High to Low ○

Price - Low to High ○

ITEM CONDITION

Condition Choose

PRICE

Minimum

Audio | Computers | Computer Parts & Accessories | Others

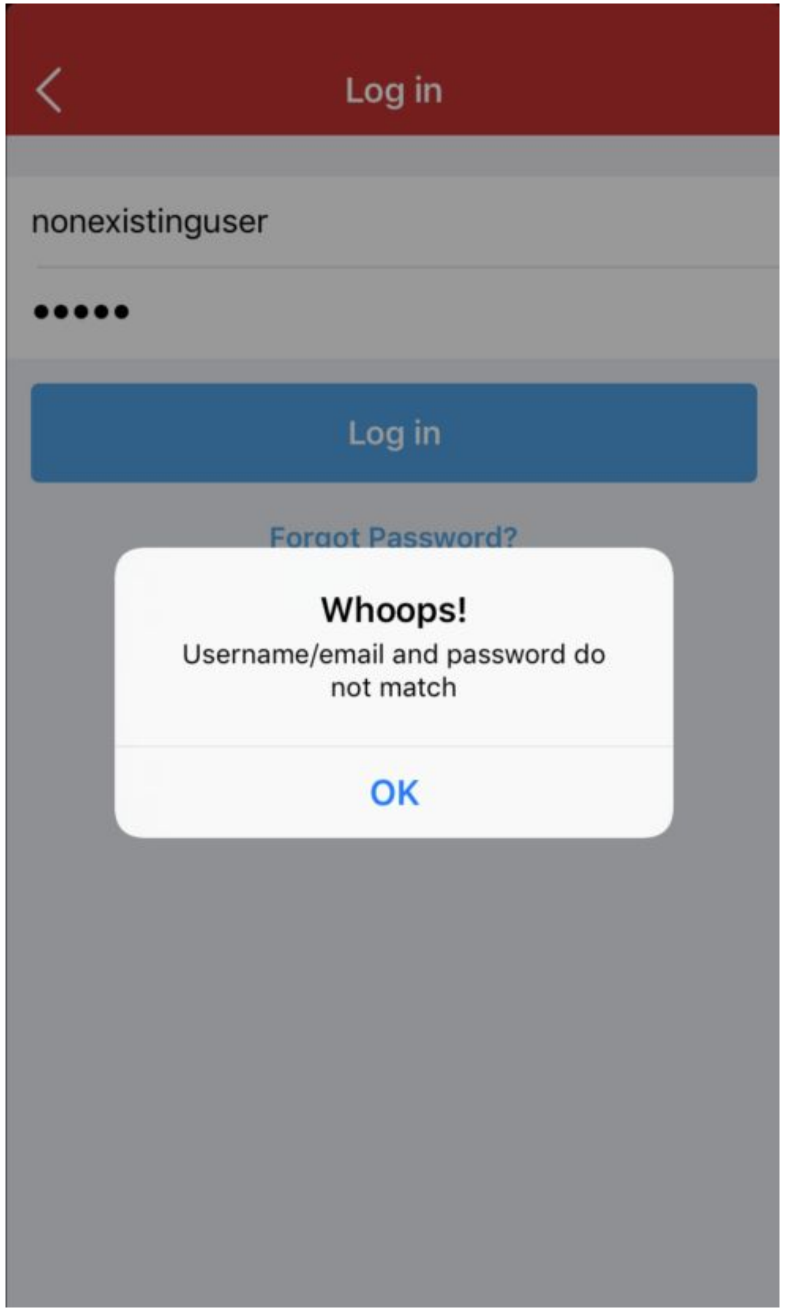
diakoh, 4 days ago	flowerbt, 18 minutes ago	sgprimeauction, 16 hours ago	hslimsg, 28 days ago
BN TOKIDOKI EZ-LI...	Creative SxFI AMP	677. FlySky 2 Chan...	Philips HTS7140 So...
S$55	S$150	S$40	S$168
ITEM DESCRIPTION Bran...	Used but in excellent con...	Atomik RC is proud to intr...	Fully functioning Soundba...
New	Used	New	Used
♡ 13	♡ 1	♡ 2	♡ 14

Multi-platform applications contd.

- Creator's motivation = maximise market coverage (and revenue)

- Development setup
  - develop for all platforms in the same functional team *or*
  - different teams for iOS, Android and Web
  - outsource development for certain platforms
  - different maturity level on different platforms
  - different feature sets on different platforms
  - …

- **BUT user expectation = same UI/UX across all platforms**



Log in

nonexistinguser

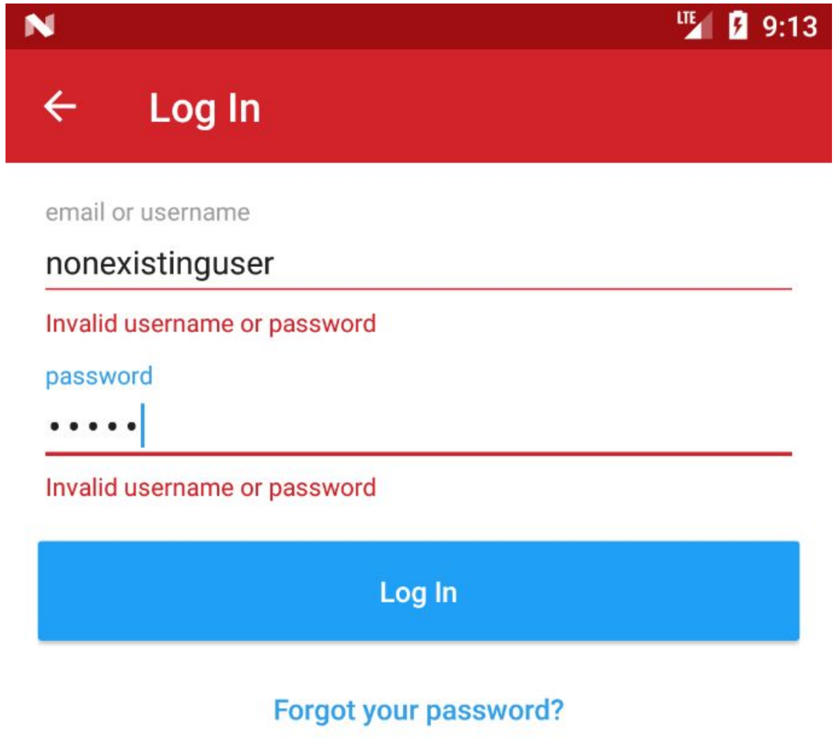
●●●●●

Log in

Forgot Password?

Whoops!
Username/email and password do not match

OK



9:13

Log In

email or username

nonexistinguser

Invalid username or password

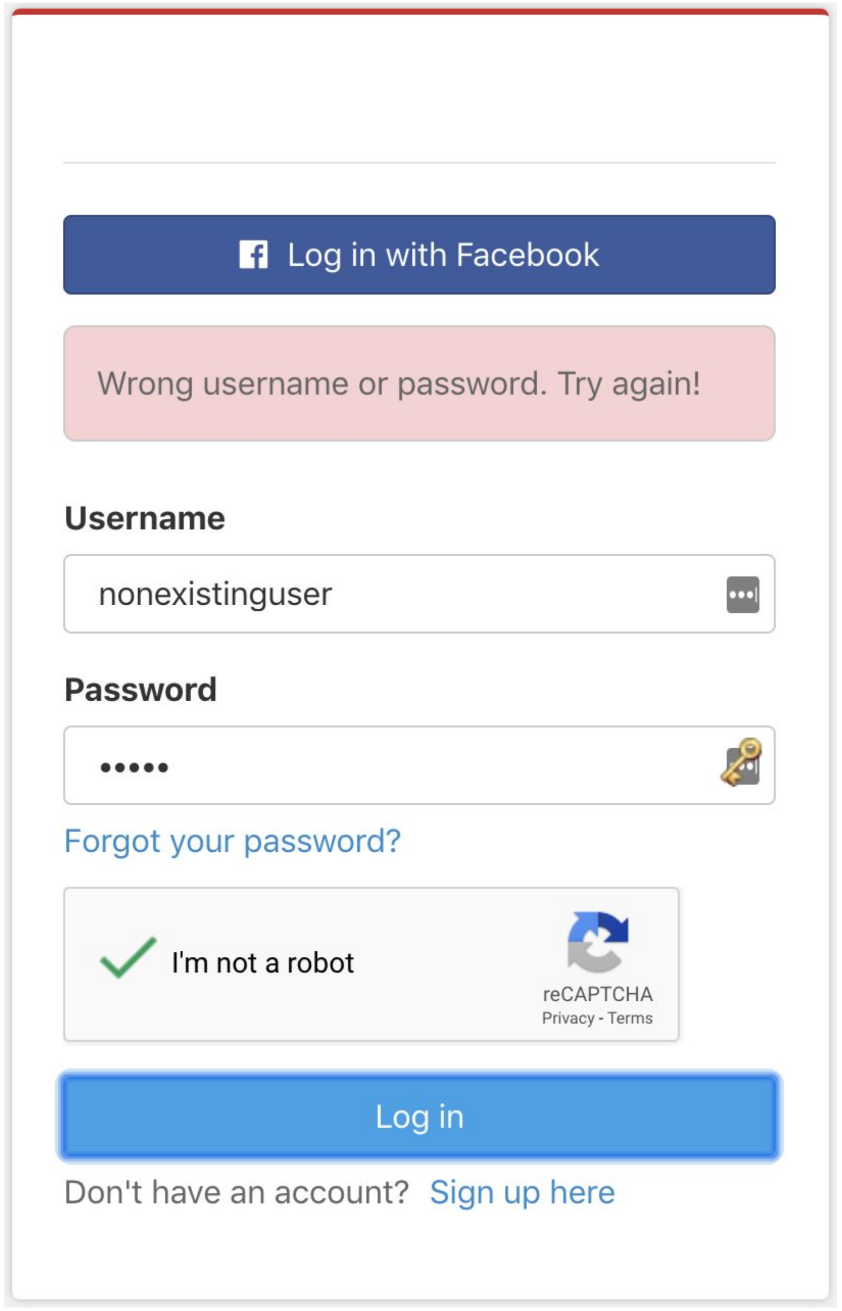
password

•••••

Invalid username or password

Log In

Forgot your password?



Log in with Facebook

Wrong username or password. Try again!

Username

nonexistinguser

Password

•••••

Forgot your password?

I'm not a robot

reCAPTCHA
Privacy - Terms

Log in

Don't have an account? Sign up here

Testing each platform separately?

aka “the naive way to multi-platform testing”

- different (but optimised) frameworks
- different test-cases (while actually testing the same)
- different integrations into the build pipeline
- different QA teams
- different stakeholders
- …

WHAT IF I TOLD YOU
WE CAN DO BETTER
memegenerator.net

Parts of UI automation testing

- Test definitions (test cases, features, scenarios…)
- Test data
- UI model (page objects, locators)
- Test framework
- Test code
- Test devices
- Test reports
- Test infrastructure
- Test engineers

How much can we re-use across platforms?

Parts of UI automation testing

Is it platform independent?

- Test definitions **YES**
- Test data **YES**
- Page objects **MOSTLY**
- Test framework **ABSOLUTELY**
- Test code **YES**
- Test devices
- Test reports **YES**
- Test infrastructure **MOSTLY**
- Test engineers **YES** 😊

SHOW ME WHAT YOU GOT

One framework to test them all

Tech-stack

- Cucumber
- Java
- Selenium
- Appium
- Spring
- Maven
- ...

This is one implementation of an idea. The concept equally applies to other technologies!

Toolbox

Selenium was first released in 2004 by Jason Huggins at ThoughtWorks
WebDriver was created in 2007 by Simon Stewart (ThoughtWorks)
→ merged into Selenium 2.0 in 2009

Appium was first released in 2011 by Dan Cuellar, since 2013 funded by SauceLabs

Cucumber was first released in 2008 by Aslak Hellesøy (guess where ;-)

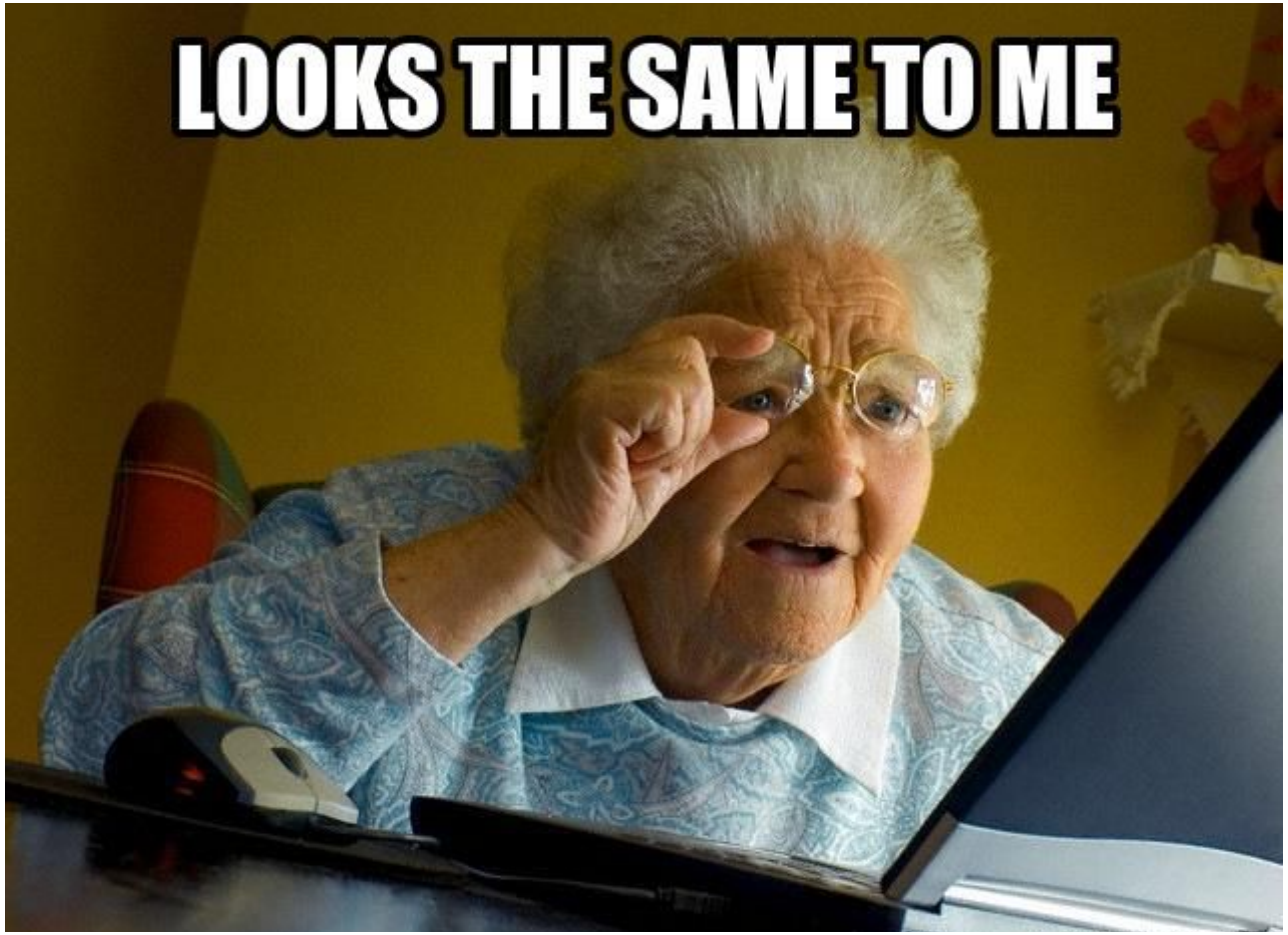
There are Java versions for all of them!

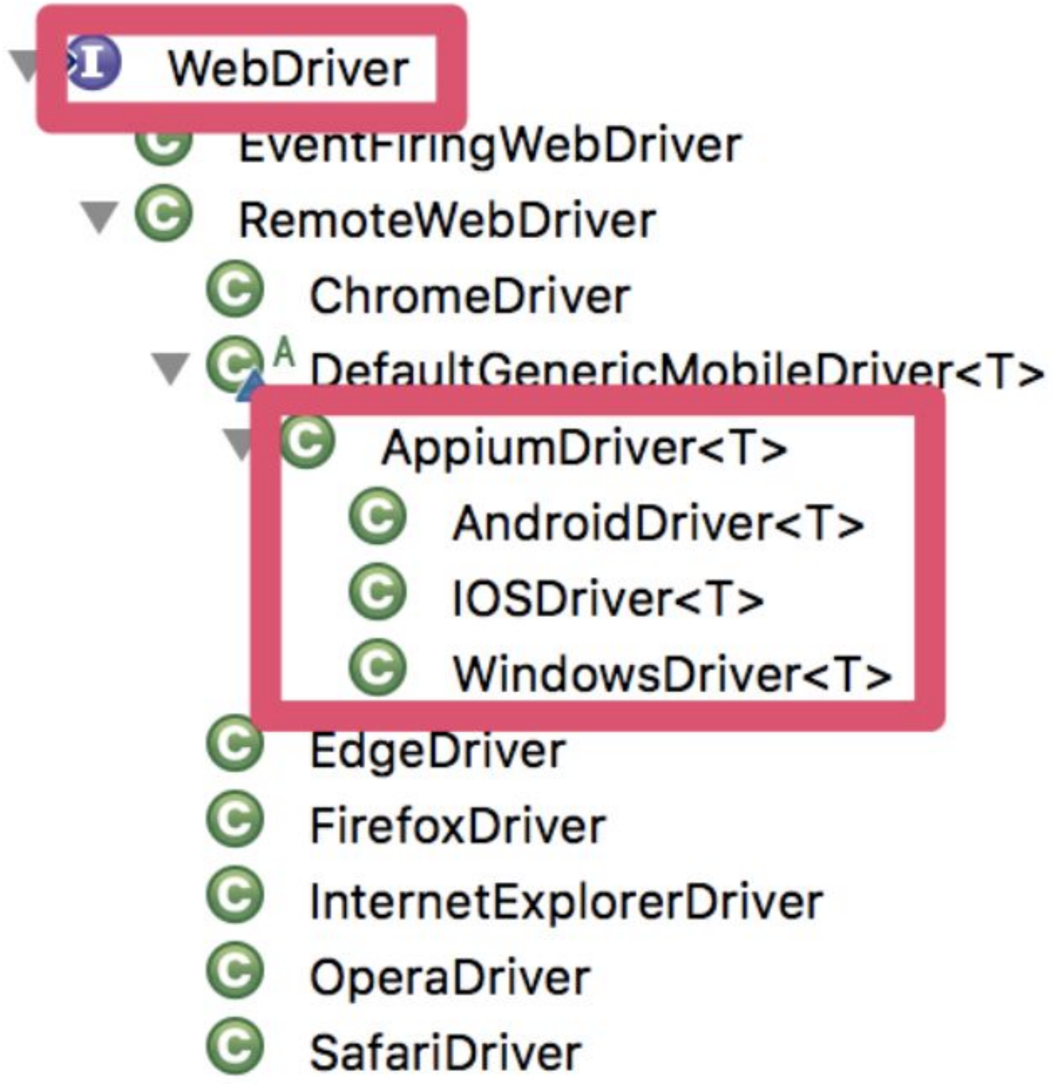
Selenium

```
WebDriver driver = new FirefoxDriver();
driver.get("http://www.google.com");

WebElement element =
driver.findElement(By.name("q"));
element.sendKeys("Manilla is awesome");
element.submit();
...
```

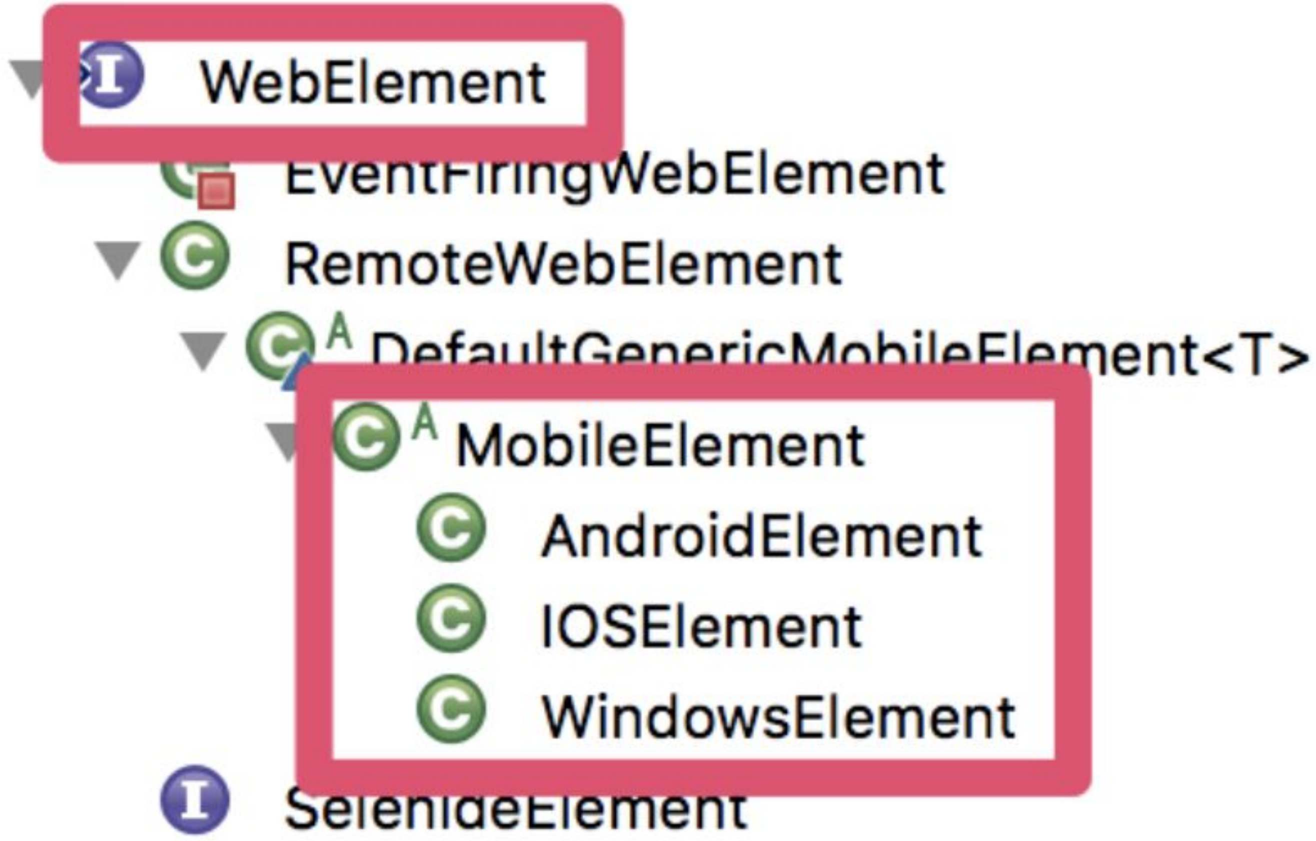
Appium

```
AppiumDriver driver = new AppiumDriver(new
URL("http://127.0.0.1:4723/wd/hub"), capabilities);

IOSElement element =
driver.findElement(By.name("search"));
element.sendKeys("Manilla is awesome");
element.submit();
...
```


LOOKS THE SAME TO ME



```
WebDriver
    EventFiringWebDriver
    RemoteWebDriver
        ChromeDriver
        DefaultGenericMobileDriver<T>
            AppiumDriver<T>
                AndroidDriver<T>
                IOSDriver<T>
                WindowsDriver<T>
        EdgeDriver
        FirefoxDriver
        InternetExplorerDriver
        OperaDriver
        SafariDriver
```



```
WebElement
    EventFiringWebElement
    RemoteWebElement
        DefaultGenericMobileElement<T>
            MobileElement
                AndroidElement
                IOSElement
                WindowsElement
    SelenideElement
```

Feature files

- Feature files are a human-readable description of use-cases
- Some features may only be available on certain platforms
- BUT **a vast majority of features will be the same** across all of them

```
@web @android @ios
Scenario Outline: Filter by tags
     Given I am on the home screen
     When I go to the tags screen
     And I filter for "<tag>"
     And I select the tag "<tag>"
     And I select the first question
     Then the question is tagged with "<tag>"
     Examples:
          | tag |
          | selenium |
          | appium |
```

Test code (glue)

```java
public class QuestionSteps extends BaseSteps {
  private QuestionPage questionPage;

  private QuestionsPage questionsPage;

  @Given("^I select the first question$")
  public void selectFirstQuestion() {
    questionsPage.openFirstQuestion();
  }

  @Then("the question is tagged with \"([^\"]*)\"")
  public void isQuestionTaggedWith(String tagName) {
    assertThat(questionPage.hasTag(tagName))
        .as("Check that tag " + tagName + " is present")
        .isTrue();
  }
}
```

Page objects

```java
@Component
@Profile(Platform.WEB)

public class HomePage extends
BasePage<HomePage> {

  private QuestionsPage questions;
  private TagsPage tags;

  public HomePage load() {
    open(configuration.getBaseUrl());
    return this;
  }
…
```

```java
…

public TagsPage navigateToTagsPage() {
    $("MENU_TAGS").click();
    return tags;
  }

  public QuestionsPage search(String query) {
    $("SEARCH_FIELD").sendKeys(query);
    $("SEARCH_BUTTON").should(appear).click();
    return questions;
  }
}
```

Page objects contd.

```
@Component
@Profile({Platform.ANDROID})

public class AndroidHomePage extends HomePage
{
  private QuestionsPage questionsPage;

…

  @Override
  public QuestionsPage search(String query) {
    $("SEARCH_FIELD").sendKeys(query + "\n");
    return questionsPage;
  }
…
```

Inheritance!

Element locators

```
SEARCH_FIELD:
  android:
    type: id
    value: com.stackexchange.stackoverflow:id/search_src_text
  web:
    type: css
    value: input[name=q]

SEARCH_BUTTON:
  web:
    type: css
    value: .iconSearch
```

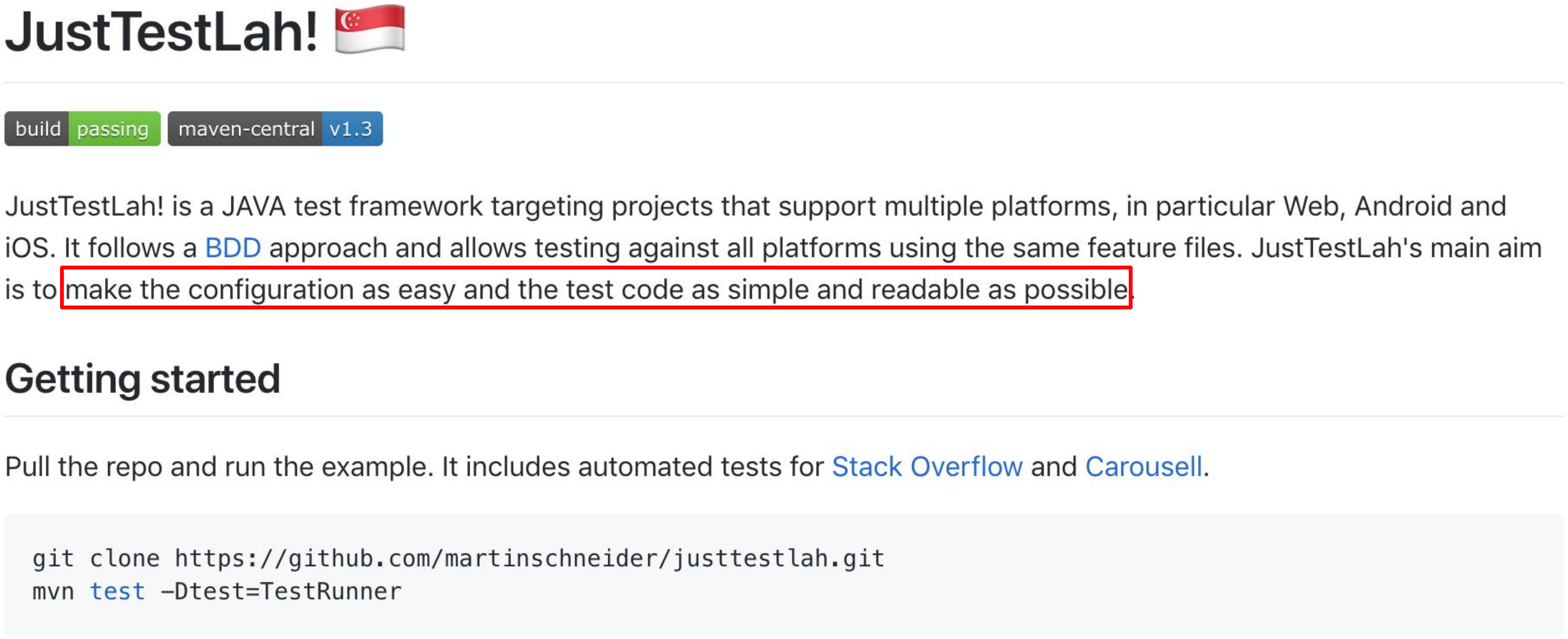
Configuration

```
# GENERAL settings
platform=web
pages.package=ph.test.pages
steps.package=ph.test.steps
features.directory=src/test/resources/features

# WEB settings
web.baseUrl=https://www.stackoverflow.com
web.browser=chrome
web.headless=true

# MOBILE settings
mobile.appiumUrl=http://127.0.0.1:4723/wd/hub
```

Configuration contd.

```
# ANDROID settings
android.appPackage=ph.test.app
android.appActivity=ph.test.MainActivity
android.appPath=/Users/martinschneider/test.apk
android.deviceName=Google Pixel

# IOS settings
ios.appPath=
ios.deviceName=iPhone 6
```

Demo #1



JustTestLah! 🇸🇬

build | passing   maven-central | v1.3

JustTestLah! is a JAVA test framework targeting projects that support multiple platforms, in particular Web, Android and iOS. It follows a BDD approach and allows testing against all platforms using the same feature files. JustTestLah's main aim is to make the configuration as easy and the test code as simple and readable as possible.

Getting started

Pull the repo and run the example. It includes automated tests for Stack Overflow and Carousell.

```
git clone https://github.com/martinschneider/justtestlah.git
mvn test -Dtest=TestRunner
```

Advantages

- All test definitions are kept in one place
  - discrepancies between platforms become visible and taken care of
  - features (= test definitions) act as an umbrella across different development teams
- A majority of the features are written only once
  - and re-used for all platforms
- The same holds for the glue code

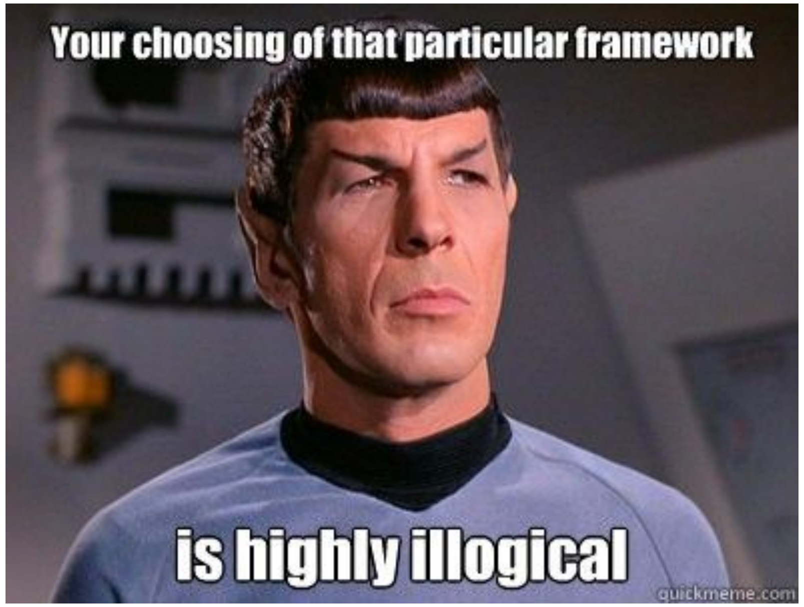
Advantages contd.

- Page objects can share code for common functionality
  - only platform-specific nuances need to be added on top
- Spring & co. take care of the heavy-lifting
  - flat learning curve for writing new test cases
- Tech-stack is well established and maintained
  - Cucumber, Selenium, Appium, Java, Spring

QA as an "umbrella"

Points to consider

- **Every framework needs to match the culture of the project**
  - for highly independent teams there might be better solutions than a common testing framework
  - however, there are still aspects worth considering: share the feature set, share the reports, learn from each other's best practices…
- BDD/TDD vs. "BDD testing"
  - the second "D" stands for development!



Points to consider contd.

- Custom framework = re-inventing the wheel?
- A custom test framework requires engineers building and maintaining it
  - tester vs. test (automation) **engineer**
- Platform-specific testing solutions might be more performant
  - "pure" Espresso, XCUITest etc.
  - weigh the pros and cons!
- What about PWAs, React Native, Flutter?

Demo #2

BrowserStack

Configuration contd.

```
cloudprovider=browserstack

# BROWSERSTACK settings
browserstack.debug=true
browserstack.username=mart.schneider@gmail.com
browserstack.accessKey=XXXYYYZZZ
```

Thank you

References

JustTestLah! test framework

- http://justtestlah.qa
- https://github.com/martinschneider/justtestlah

The making of Selenium/Cucumber/Appium

- https://www.infoq.com/news/2018/04/cucumber-bdd-ten-years
- https://www.seleniumhq.org/about/history.jsp
- http://appium.io/history.html