Testing Android Applications

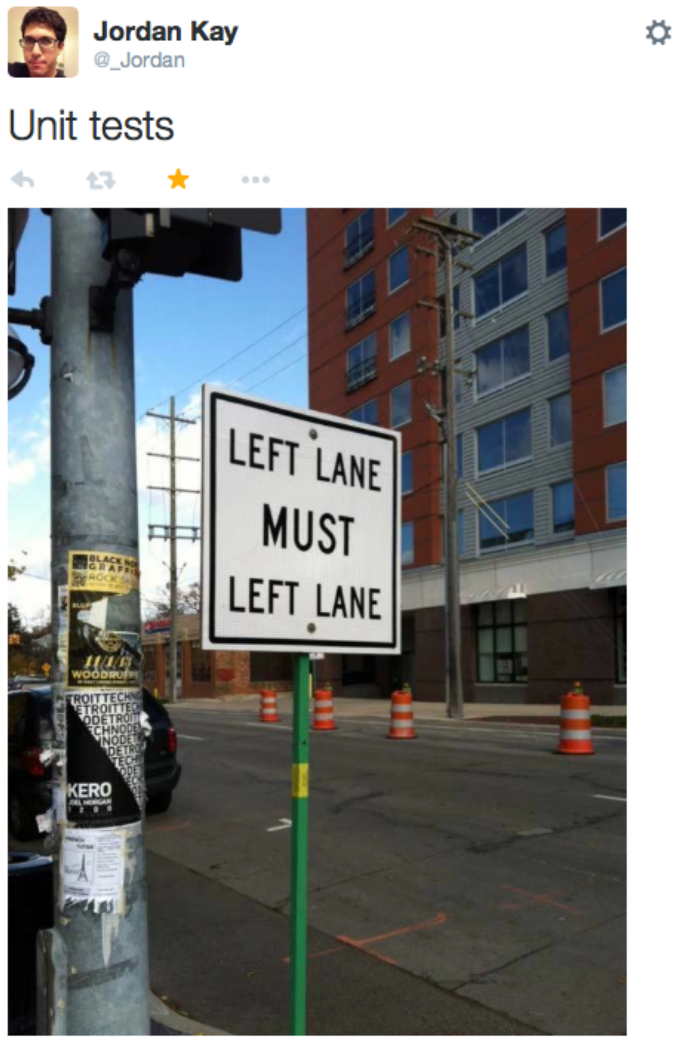
http://danosipov.com @danosipov

Agenda

- Why Test
- When and What to Test
- Unit vs Integration Testing
- Mocks and Stubs
- Robolectric and Espresso
- Demo

Why Test?

- Quality
- Faster development cycle
- Faster release cycle
- Cheaper than manual testing
- More development effort



Jordan Kay
@_Jordan

Unit tests

What to test and what not to test

- Business logic
- Behavior
- Bugs

- UX/UI
- Configuration changes
- Network state
- Implementation details

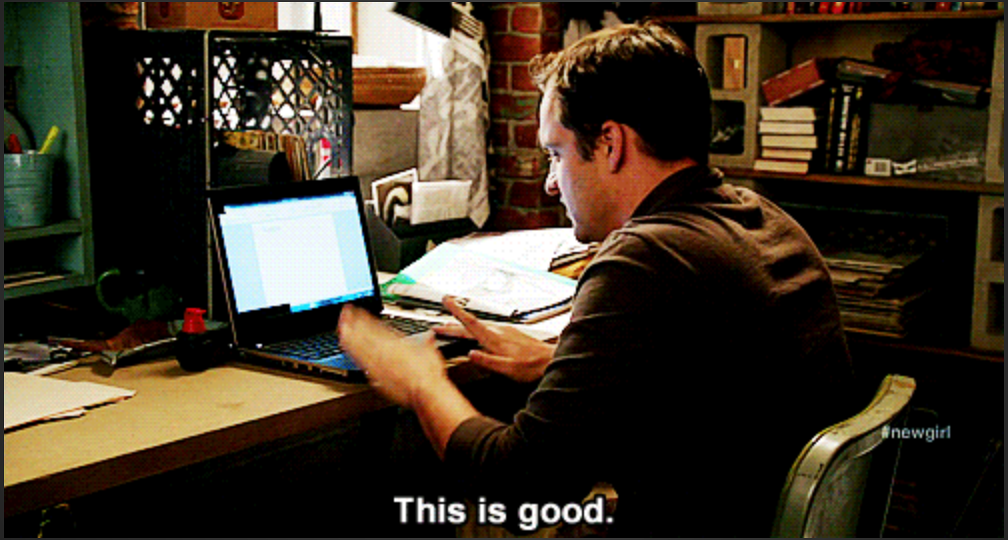
Unit Testing

- Robolectric / JUnit
- Test small pieces of functionality in isolation
  - ex: individual methods
- Fast execution time
  - Runs on the JVM, no device required
  - Stubs used for Android APIs

Integration (functional) Testing

- Espresso
- Test full application
  - Runs on the device
- Test integration with other services

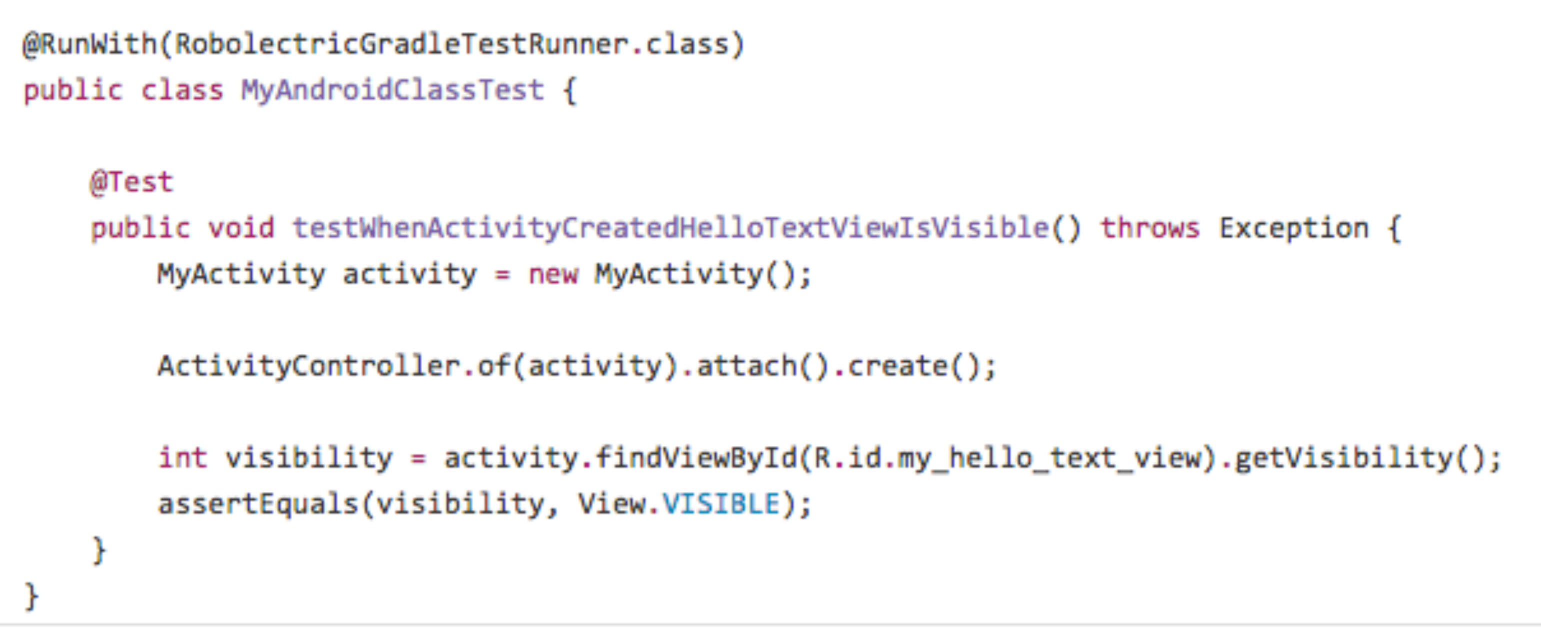
Writing tests



Test Structure

1. Arrange
2. Act
3. Assert

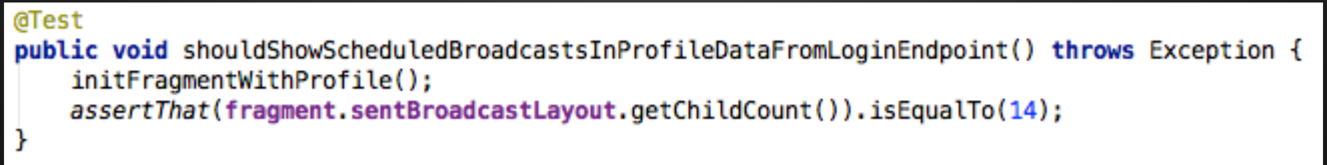
Example



```
@RunWith(RobolectricGradleTestRunner.class)
public class MyAndroidClassTest {

    @Test
    public void testWhenActivityCreatedHelloTextViewIsVisible() throws Exception {
        MyActivity activity = new MyActivity();

        ActivityController.of(activity).attach().create();

        int visibility = activity.findViewById(R.id.my_hello_text_view).getVisibility();
        assertEquals(visibility, View.VISIBLE);
    }
}
```

Name Tests appropriately

- Should instantly convey information about what the test expected to happen.
- Great for when the test fails.



```
@Test
public void shouldShowScheduledBroadcastsInProfileDataFromLoginEndpoint() throws Exception {
    initFragmentWithProfile();
    assertThat(fragment.sentBroadcastLayout.getChildCount()).isEqualTo(14);
}
```

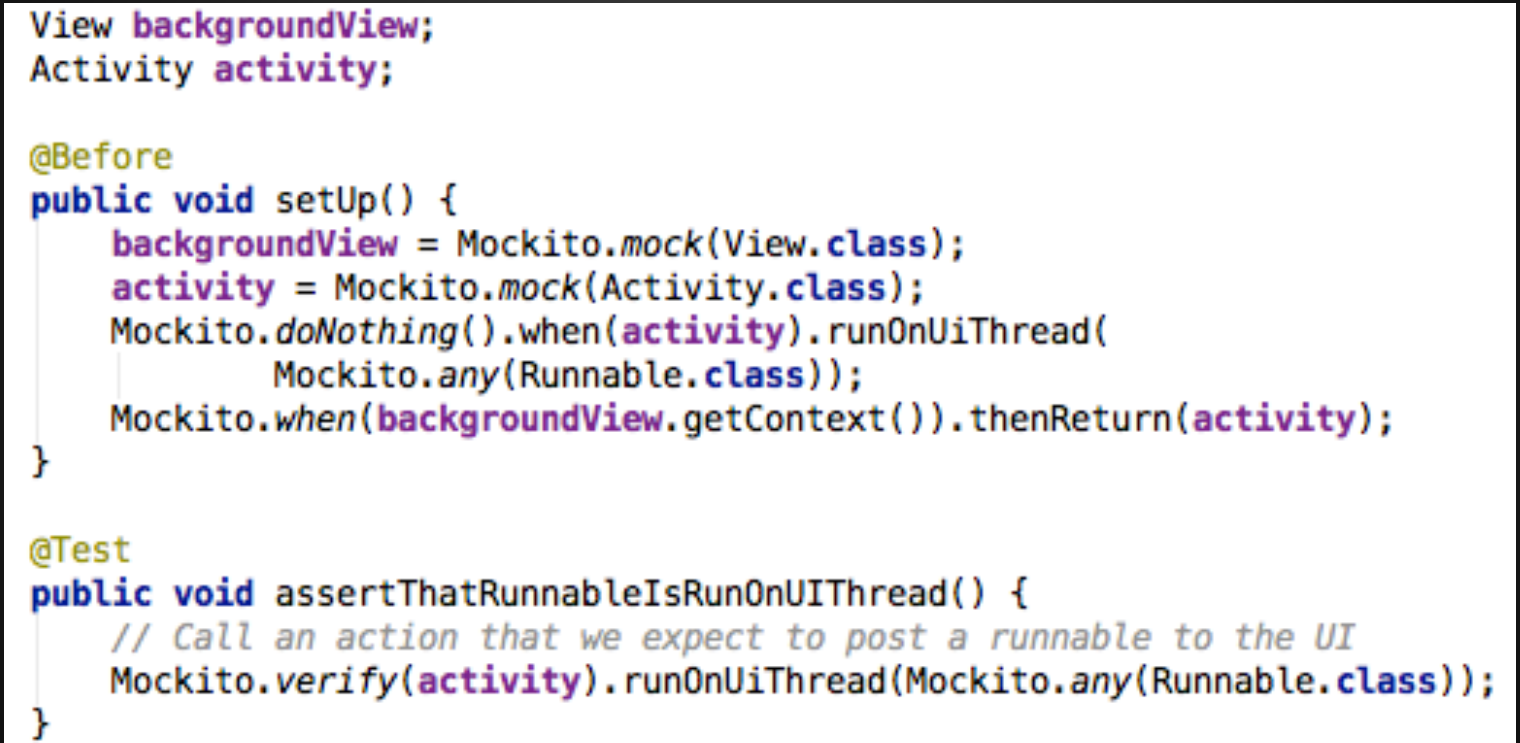
Test Suites

- Unit test classes correspond 1-to-1 to classes
- Integration classes correspond to activities/fragments

Mocks

- Mockito
- when(): Provide expected behavior
- verify(): perform assertions

Using Mockito



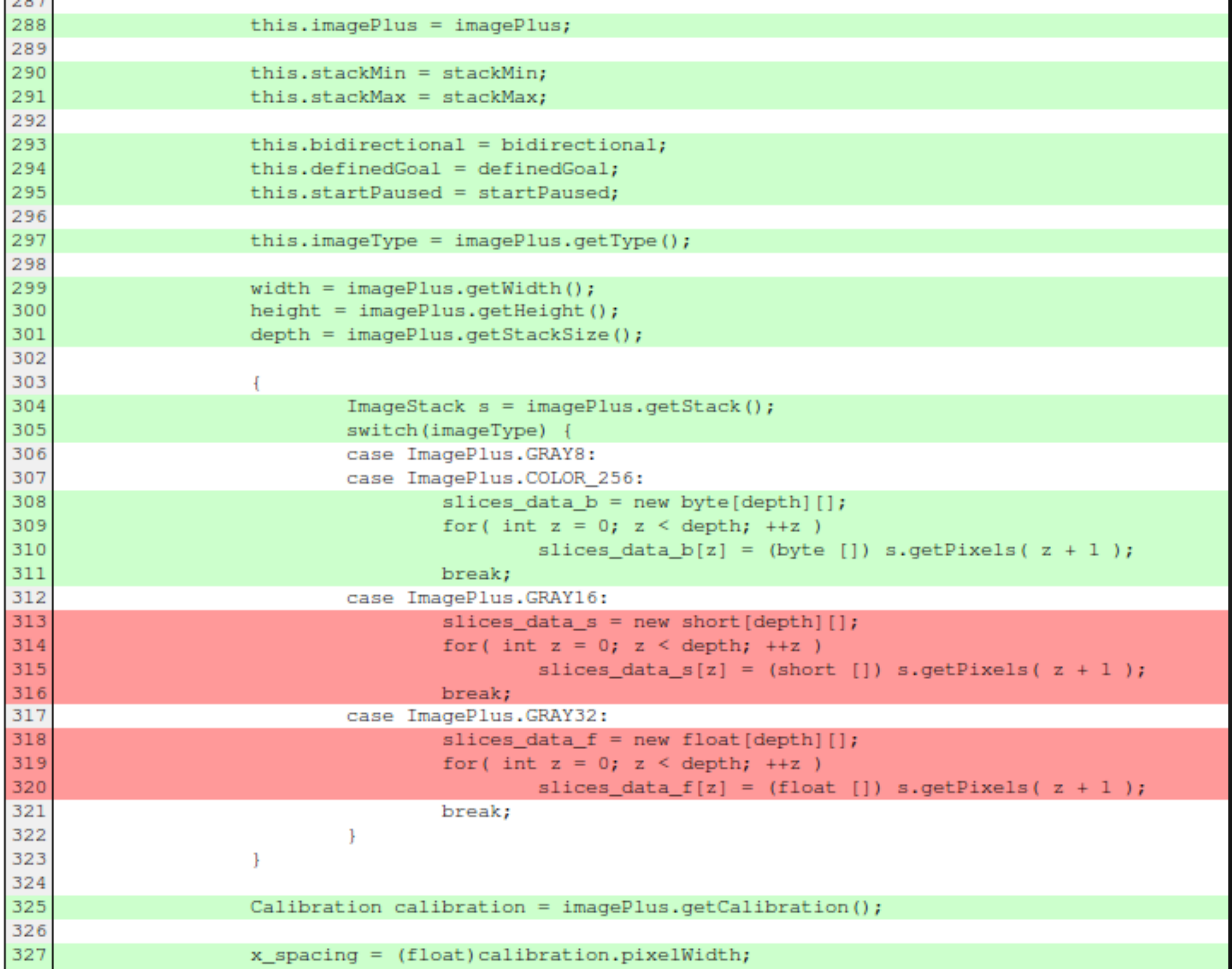
```
View backgroundView;
Activity activity;

@Before
public void setUp() {
    backgroundView = Mockito.mock(View.class);
    activity = Mockito.mock(Activity.class);
    Mockito.doNothing().when(activity).runOnUiThread(
            Mockito.any(Runnable.class));
    Mockito.when(backgroundView.getContext()).thenReturn(activity);
}

@Test
public void assertThatRunnableIsRunOnUIThread() {
    // Call an action that we expect to post a runnable to the UI
    Mockito.verify(activity).runOnUiThread(Mockito.any(Runnable.class));
}
```

Stubs

- Like Mocks, but you provide an implementation.
- Works well with interfaces
- Ex: Robolectric

DEMO!

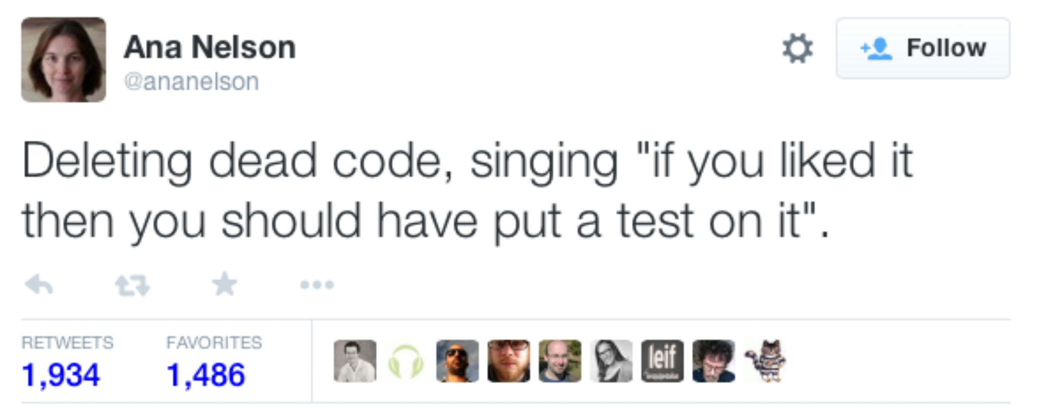
ANDROID



```
                this.imagePlus = imagePlus;

                this.stackMin = stackMin;
                this.stackMax = stackMax;

                this.bidirectional = bidirectional;
                this.definedGoal = definedGoal;
                this.startPaused = startPaused;

                this.imageType = imagePlus.getType();

                width = imagePlus.getWidth();
                height = imagePlus.getHeight();
                depth = imagePlus.getStackSize();

                {
                        ImageStack s = imagePlus.getStack();
                        switch(imageType) {
                        case ImagePlus.GRAY8:
                        case ImagePlus.COLOR_256:
                                slices_data_b = new byte[depth][];
                                for( int z = 0; z < depth; ++z )
                                        slices_data_b[z] = (byte []) s.getPixels( z + 1 );
                                break;
                        case ImagePlus.GRAY16:
                                slices_data_s = new short[depth][];
                                for( int z = 0; z < depth; ++z )
                                        slices_data_s[z] = (short []) s.getPixels( z + 1 );
                                break;
                        case ImagePlus.GRAY32:
                                slices_data_f = new float[depth][];
                                for( int z = 0; z < depth; ++z )
                                        slices_data_f[z] = (float []) s.getPixels( z + 1 );
                                break;
                        }
                }

                Calibration calibration = imagePlus.getCalibration();

                x_spacing = (float)calibration.pixelWidth;
```

Testing culture

- Discipline
- Code review
- Test Driven Development

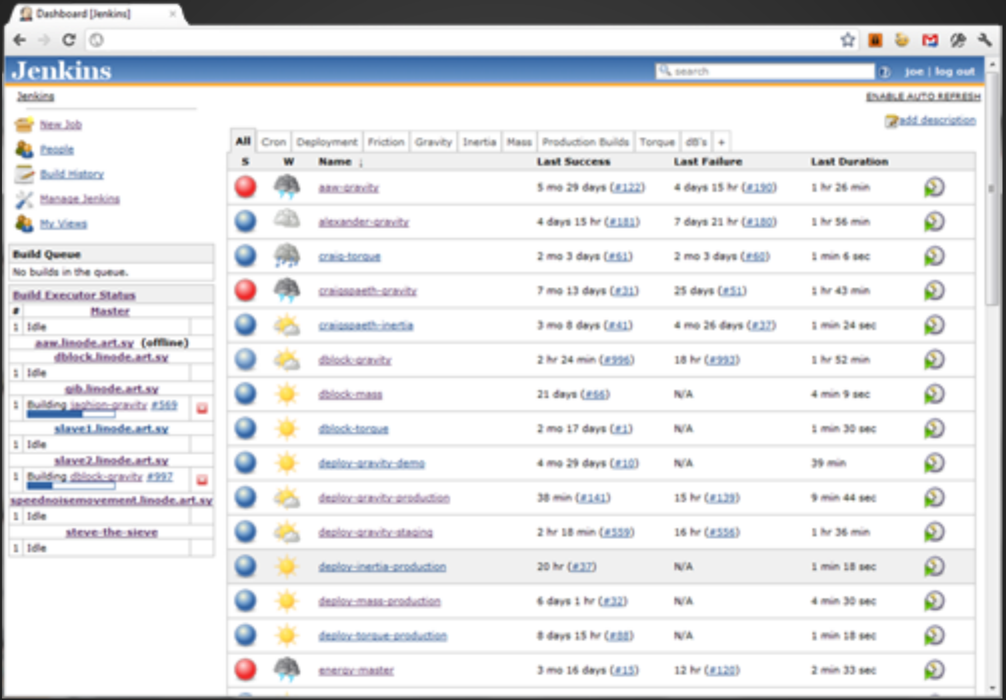


Ana Nelson
@ananelson

Deleting dead code, singing "if you liked it then you should have put a test on it".

RETWEETS 1,934 FAVORITES 1,486

Jenkins CI



Dashboard [Jenkins]

Jenkins search joe | log out

Jenkins

ENABLE AUTO REFRESH

- New Job
- People
- Build History
- Manage Jenkins
- My Views

add description

Build Queue

No builds in the queue.

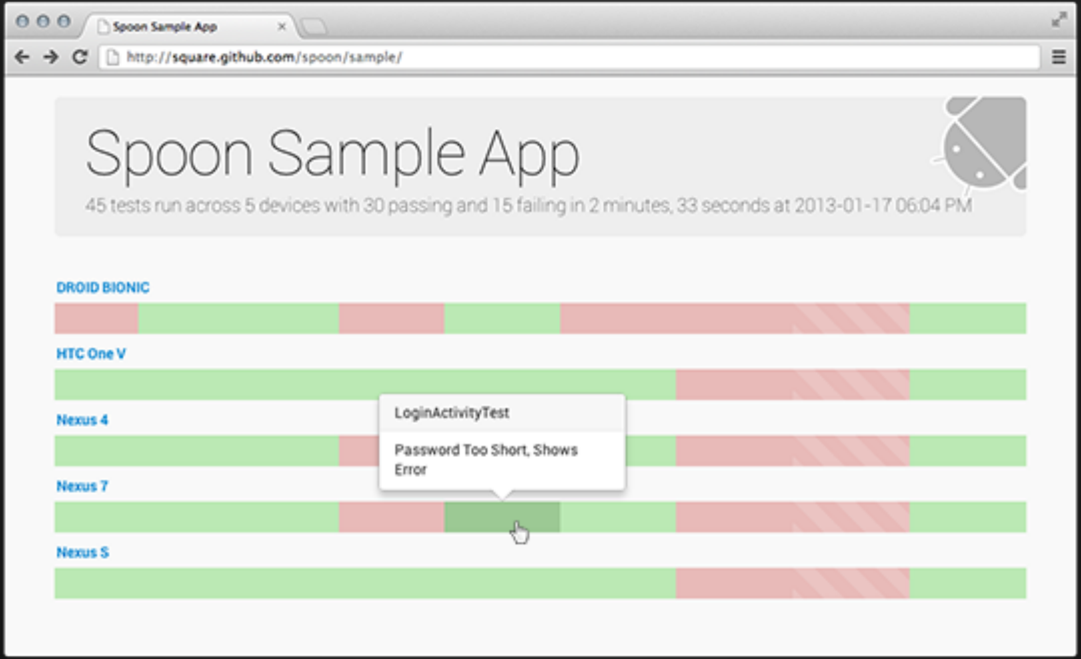
Build Executor Status

#	Master	
1	Idle	
	aaw.linode.art.sy (offline)	
	dblock.linode.art.sy	
1	Idle	
	gib.linode.art.sy	
1	Building fashion-gravity #569	☒
	slave1.linode.art.sy	
1	Idle	
	slave2.linode.art.sy	
1	Building dblock-gravity #997	☒
	speednoisemovement.linode.art.sy	
1	Idle	
	steve-the-sieve	
1	Idle	

All | Cron | Deployment | Friction | Gravity | Inertia | Mass | Production Builds | Torque | dB's | +

S	W	Name ↓	Last Success	Last Failure	Last Duration	
🔴	🌧	aaw-gravity	5 mo 29 days (#122)	4 days 15 hr (#190)	1 hr 26 min	
🔵	☁	alexander-gravity	4 days 15 hr (#181)	7 days 21 hr (#180)	1 hr 56 min	
🔵	🌧	craig-torque	2 mo 3 days (#61)	2 mo 3 days (#60)	1 min 6 sec	
🔴	🌧	craigspaeth-gravity	7 mo 13 days (#31)	25 days (#51)	1 hr 43 min	
🔵	⛅	craigspaeth-inertia	3 mo 8 days (#41)	4 mo 26 days (#37)	1 min 24 sec	
🔵	⛅	dblock-gravity	2 hr 24 min (#996)	18 hr (#993)	1 hr 52 min	
🔵	☀	dblock-mass	21 days (#66)	N/A	4 min 9 sec	
🔵	☀	dblock-torque	2 mo 17 days (#1)	N/A	1 min 30 sec	
🔵	☀	deploy-gravity-demo	4 mo 29 days (#10)	N/A	39 min	
🔵	⛅	deploy-gravity-production	38 min (#141)	15 hr (#139)	9 min 44 sec	
🔵	⛅	deploy-gravity-staging	2 hr 18 min (#559)	16 hr (#556)	1 hr 36 min	
🔵	☀	deploy-inertia-production	20 hr (#37)	N/A	1 min 18 sec	
🔵	☀	deploy-mass-production	6 days 1 hr (#32)	N/A	4 min 30 sec	
🔵	☀	deploy-torque-production	8 days 15 hr (#88)	N/A	1 min 18 sec	
🔴	🌧	energy-master	3 mo 16 days (#15)	12 hr (#120)	2 min 33 sec	

Spoon



Spoon Sample App

http://square.github.com/spoon/sample/

Spoon Sample App

45 tests run across 5 devices with 30 passing and 15 failing in 2 minutes, 33 seconds at 2013-01-17 06:04 PM

DROID BIONIC

HTC One V

Nexus 4

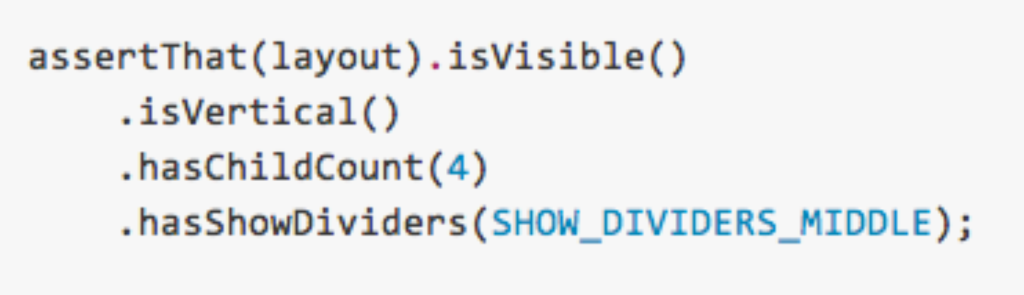
Nexus 7

Nexus S

LoginActivityTest

Password Too Short, Shows Error

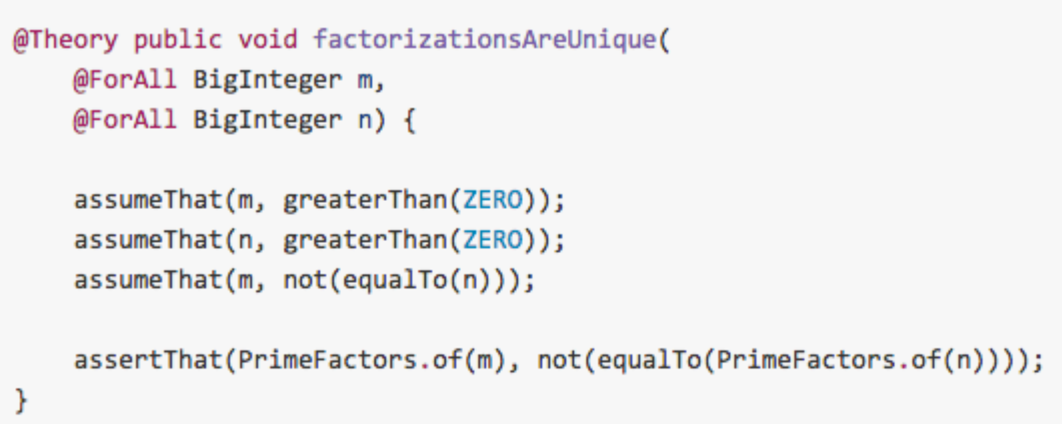
AssertJ Android



```
assertThat(layout).isVisible()
    .isVertical()
    .hasChildCount(4)
    .hasShowDividers(SHOW_DIVIDERS_MIDDLE);
```

Property Testing

Quickcheck



```
@Theory public void factorizationsAreUnique(
    @ForAll BigInteger m,
    @ForAll BigInteger n) {

    assumeThat(m, greaterThan(ZERO));
    assumeThat(n, greaterThan(ZERO));
    assumeThat(m, not(equalTo(n)));

    assertThat(PrimeFactors.of(m), not(equalTo(PrimeFactors.of(n))));
}
```

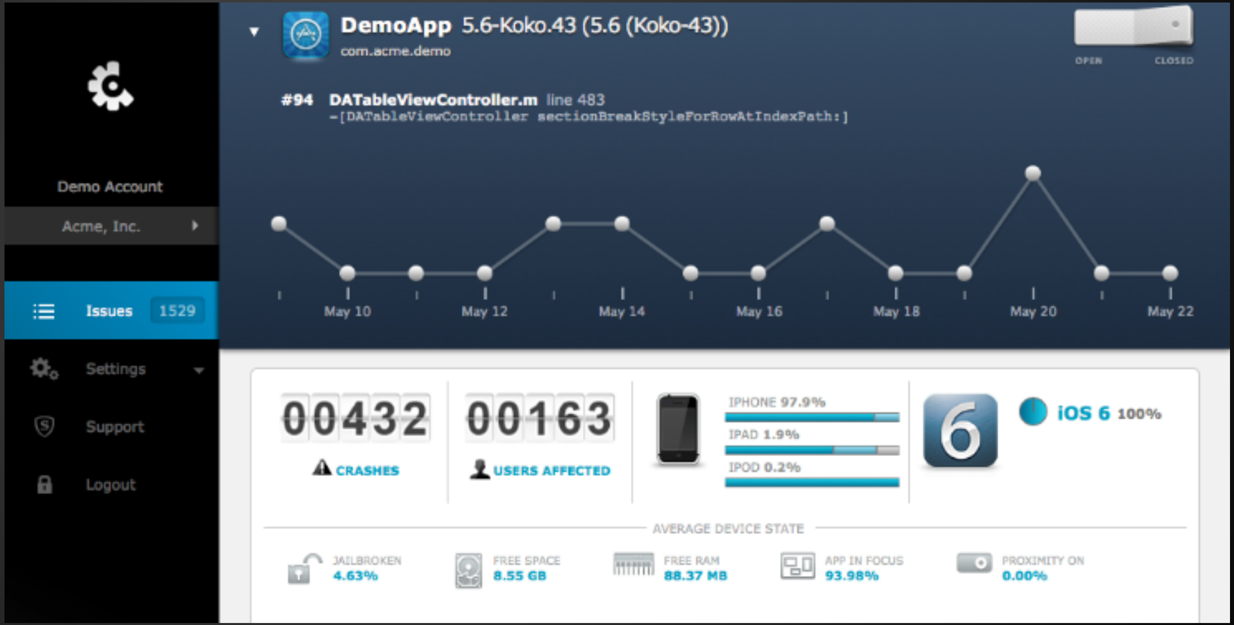
We’ll do it LIVE!

- Dogfooding
- Dark Features
- AB Testing

I DON'T ALWAYS TEST MY CODE
BUT WHEN I DO I DO IT IN PRODUCTION
memegenerator.net

Monitoring & Distribution

- Crashlytics
- Hockeyapp



DemoApp 5.6-Koko.43 (5.6 (Koko-43))
com.acme.demo

#94 DATableViewController.m line 483

```
-[DATableViewController sectionBreakStyleForRowAtIndexPath:]
```

00432 CRASHES

00163 USERS AFFECTED

IPHONE 97.9%
IPAD 1.9%
IPOD 0.2%

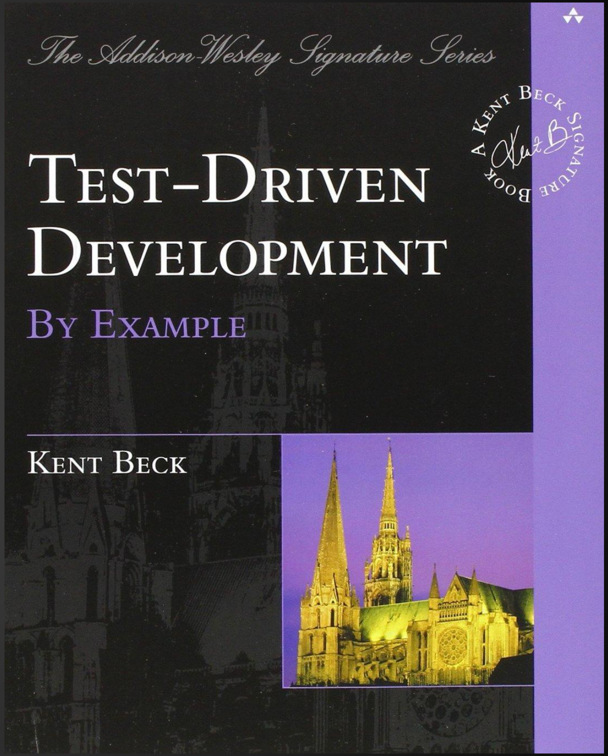
iOS 6 100%

AVERAGE DEVICE STATE

JAILBROKEN 4.63%
FREE SPACE 8.55 GB
FREE RAM 88.37 MB
APP IN FOCUS 93.98%
PROXIMITY ON 0.00%

Resources

- Robolectric Guide
- Espresso Guide


The Addison-Wesley Signature Series
A Kent Beck Signature Book
Test-Driven Development
By Example
Kent Beck

Questions?