

Demystifying Android Testing



Jon Holtan
Android Engineer
Tribalscale

TribalScale

- ✦ Our pride is in creating digital partnerships.
- ✦ Our power is in our people.
- ✦ Our locations are global.
- ✦ We are [hiring](#) for all positions and locations.

Unit Testing Defined

- ✿ Unit testing is a software testing method, in which small units of code are tested. - [Wikipedia](#)
- ✿ Unit tests are the fundamental tests in your app testing strategy. - [Google](#)
- ✿ We're all programmers here and we we can automate these things. - [Jake Wharton](#)

Myths you may have heard...

- ✦ Android testing is impossible.
- ✦ Testing isn't for me.
- ✦ It takes too long.

Truths you need to hear...

- ✦ Overall Timesaver > Initial Setup Time
- ✦ Well written tests back you up
- ✦ Anything can be tested with the right strategy
- ✦ Decreases time spent debugging

My Testing Strategy

- ✦ Break up your code, your classes, your functions.
- ✦ When in doubt **mock** it.
- ✦ Can't mock, but still need to verify? Then **spy**.
- ✦ Hard time testing? Maybe it is time for a refactor.

Mocks and Spies? Am I an assassin?

Yes!

A bug and regression
assassinator!

In all seriousness: Some terms

- ✦ **Mock:** A fake copy of your class with empty methods.
- ✦ **Spy:** A real copy of your class with all functionality in tact.
- ✦ <http://www.baeldung.com/mockito-spy>

Mockito

- ✦ A mocking framework that tastes really good.
- ✦ The best mocking framework for Java. - [StackOverflow](#)
- ✦ “We decided that we should use JUnit 4 and Mockito because we think they are the future of TDD and mocking in Java” - [Dan North](#)
- ✦ <http://site.mockito.org/>

Mockito: How?

```
public class MainActivityPresenterTest {  
    @Mock  
    MainActivityContract.View mockView;  
  
    MainActivityPresenter presenter;  
  
    @Before  
    public void setUp() throws Exception {  
        MockitoAnnotations.initMocks(this);  
  
        presenter = new MainActivityPresenter();  
    }  
  
    @Test  
    public void createShouldSetAView() throws Exception {  
        assertNull(presenter.view);  
  
        presenter.create(mockView);  
  
        assertNotNull(presenter.view);  
    }  
}
```

Robolectric

- ✦ Tests run inside the JVM on your workstation in seconds.
- ✦ De-fangs the Android SDK jar.
- ✦ Inflates Views, Loads Resources, and handles other native “stuff”
- ✦ Works with mocking frameworks (Mockito)
- ✦ <http://robolectric.org/>

Robolectric: What?

```
@RunWith(RobolectricTestRunner.class)
@Config(constants = BuildConfig.class)
public class LauncherActivityTest {

    LauncherActivity launcherActivity;

    @Before
    public void setUp() throws Exception {
        launcherActivity = Robolectric.buildActivity(LauncherActivity.class).get();
    }

    @Test
    public void shouldLaunchMainActivity() throws Exception {
        launcherActivity.onLaunchClick();

        Intent launchedActivity =
            shadowOf(launcherActivity).getNextStartedActivity();

        assertNotNull(launchedActivity);
        assertEquals(
            MainActivity.class.getCanonicalName(),
            launchedActivity.getComponent().getClassName()
        );
    }
}
```

Demo Time!!!

