# Software Engineering Large Practical: Testing Android applications

Stephen Gilmore School of Informatics Wednesday 1<sup>st</sup> November, 2017

- 1. Software testing
- 2. Recording user interface tests
- 3. Running tests

## Software testing

- In practice, software is produced to tight product deadlines. Changes to product requirements are frequent, often because our software is interacting with other software that is changing.
- The principal method of improving software quality is through automated testing, with test frameworks being used to encode tests which can be re-run every time that the code is updated.
- We will only be introducing the very important topic of software testing here, and only specifically for Android applications. For an in-depth look at the subject see the Software Testing course next semester.

### Expectations of software testing

Program testing can be used to show the presence of bugs, but never to show their absence!

— Edsger W. Dijkstra, 1970.

- In 2017, very few people expect to be able to prove that their code has no bugs. Perfection in software development is not widely viewed as an attainable goal.
- However, user-visible errors in software cause reputational damage and can cause your app to be uninstalled or receive negative user reviews.
- Pragmatically, the goal of software testing is to reduce the number of bugs in code which is shipped to the user.

### Needs more testing



- Android supports two types of testing: unit tests and instrumented tests.
- Unit tests are located under src/test/java, run on the JVM, and do not have access to Android APIs.
- Instrumented tests go under src/androidTest/java, run on a hardware device or the emulator, and can invoke methods and modify fields in your application.
- Both types of tests are valuable, but here we will focus on instrumented tests, in particular user interface tests using the Espresso test framework.

https://developer.android.com/training/testing/fundamentals.html

### Unit tests and instrumented tests



http://developer.android.com/training/testing/start/

## **Recording user interface tests**

### **Recording user interface tests**

• Android Studio provides the Espresso Test Recorder which tracks our interactions with our app while it is being used, and generates an Espresso test to replay these interactions in automated testing later.

## Starting the Espresso test recorder

de Analyze	Refactor	Build	Run Tools VCS Window Help			
J MapsActi	ivity.java - S	Songnam	Run 'MapsActivityTest'	^ R	nbulist]	
psActivityTest -	▶ 4 菲	ı⊾ 🖬	Apply Changes: No active 'MapsActivity	/Test' launch		
ac ad Elinf	- Conon	ambuliet	Debug 'MapsActivityTest'	^D		
	a soligi	ambunse	Run 'MapsActivityTest' with Coverage			
0 ÷ 14+ 1	C MapsA	ctivity.jav.	Run	^7CR	ivity.java × C SongsDbHeiper.jav	a x Currentso
		MapsAct	🎼 Debug	^\CD		
	1	package	Record Espresso Test			
	2	Annant	Edit Configurations			
	4	import	Import Test Results	•		
	5	import	Apply Changes: No active 'MansActivity	Test' launch ¥E10		
	6	import	Cton	9850		
	7	import	Chan Duration List	00 F.Z		
	9	import	Snow Running List			
	10	import	Restart Activity			
	11	import	- Stop Over	50		
	12	import	Eares Step Over	7- 4-50		
	14	Linport	Porce Step Over	7. 11-18		
	15	import	Step Into	F7		
	16	import	🤰 Force Step Into	℃ 企 F7		
	17	import	验 Smart Step Into	①F7		
	19	import	🛃 Step Out	<b>①</b> F8		
	20	import	* Run to Cursor	7C F9		
	21	import	*r Force Run to Cursor	<b>乙第F9</b>		
	22	import	The Drop Frame			
	24	import	Pause Program			
			Paguma Dragram	7-920		
		_	Resulte Program	/_ d6 F(		
numbureasereese			Evaluate Expression	\_ F8		
snell pm instal	l −r "/dat	a/local	Quick Evaluate Expression	\_ ₩F8		
ſ			E Show Execution Point	\_ F10		
tests			Toggle Line Breakpoint	Жға		

© ÷ ∳∗ ⊩ ©	MapsActivity.java ×	🕑 MapsActivityTest.java 🗙	C ShakeDetector.java ×	C SettingsActivity.java ×	€ SongsDbHelper.java ×	CurrentSong.java x
	MapsActivit	ty readDatabase()				
		Record Your Test				
b) a b b b b b b b b b b b b b b b b b b		No events recorded	yet.			
¥						
rapp ing/Courses/SELP/Songnambu L/tmp/uk.ac.ed.inf.songnam			Add Assertio	n tmp/uk.ac.ed.inf.s	songnambulist	
<pre>songnambulist/uk.ac.ed.ir : uk.ac.ed.inf.songnambu' : uk.ac.ed.inf.songnambu' : uk.ac.ed.inf.songnambu' ist 'localhost:13947', transpo</pre>	? rt: 'socket'		Cancel OK	oid.intent.catego	ry.LAUNCHER -D	

⊕ ≑ ∳- ⊩	C MapsActivity.java ×	🕑 MapsActivityTest.java 🗙	C ShakeDetector.java ×	C SettingsActivity.java ×	€ SongsDbHelper.java ×	CurrentSong.java x
	MapsActivi	ty readDatabase()				
	000	Record Your Tes	t			
Ð	Delay 5000 n	nilliseconds				
1 <sup>1</sup> 1 🔲						
rapp ing/Courses/SELP/Songna L/tmp/uk.ac.ed.inf.song	mbi nar		Add Assertion	n tmp/uk.ac.ed.inf.s	songnambulist	
<pre>songnambulist/uk.ac.ed : uk.ac.ed.inf.songnam : uk.ac.ed.inf.songnam : uk.ac.ed.inf.songnam ist localhost:13947', tran</pre>	.in bu' bu' ? sport: 'socket'		Cancel OK	oid.intent.catego	ry.LAUNCHER -D	

-1 -☆ ÷ ③	G MapsActivity.java ×	🕑 MapsActivityTest.java 🗙	C ShakeDetector.java ×	C SettingsActivity.java ×	C SongsDbHelper.java x	CurrentSong.java x
	MapsActivi	ity readDatabase()				
	000	Record Your Test				
	Delay 5000 r	nilliseconds				
	Tap Floating	ActionButton with ID fab				
	1					
	1:					
	1.					
t)	1					
	1					
	2					
	2:					
	2:					
	2!					
	2'					
	2!					
1 *1 🔳	_					
rapp			Add Assertion			
l/tmp/uk.ac.ed.inf.songr	nbi har			tmp/uk.ac.ed.inf.	songnambulist	
.songnambulist/uk.ac.ed.	.i.			oid.intent.catego	ry.LAUNCHER -D	
e: uk.ac.ed.inf.songnamb	u'					
e: uk.ac.ed.inf.songnamb	u' ?		Cancel OK			
'localhost:13947', trans	sport: 'socket'					

⊕ ≑ ∦- ⊩	C MapsActivity.jav	a 🗙 🔮 MapsActivityTest.java 🗙	C ShakeDetector.java ×	C SettingsActivity.java ×	C SongsDbHelper.java ×	CurrentSong.java x
	MapsAc	tivity readDatabase()				
	000	Record Your Tes	st			
	Delay 50	00 milliseconds				
	Tap Float	ingActionButton with ID fab				
	Tap Float	ingActionButton with ID fab				
	1 Delay 500	00 milliseconds				
	1:					
	1.					
t)	1					
	1					
	21					
	2:					
	2.					
	21					
	21					
1 <sup>16</sup> 1 🔳						
rapp			Add Assertio	n .		
l/tmp/uk.ac.ed.inf.song	inar			tmp/uk.ac.ed.int.	songnambulist	
songnambulist/uk.ac.ed	i. ir			oid.intent.catego	ry.LAUNCHER -D	
e: uk.ac.ed.inf.songnam	ibu'					
ist 'localhost:13947', tran	sport: 'socket'		Cancel			

© ÷ ∦• ŀ*	C Ma	psActivity.java ×	C MapsActivityTest.java ×	C ShakeDetector.java ×	C SettingsActivity.java ×	C SongsDbHelper.java ×	CurrentSong.java x
		MapsActivi	ty readDatabase()				
		0 0	Record Your Tes	st			
		Delay 5000 m	nilliseconds				
		Tap FloatingA	ctionButton with ID fab				
		Tap FloatingA	ctionButton with ID fab				
	1	Delay 5000 m	nilliseconds				
	1: 1: 1:	Tap Overflow	MenuButton with content	description More options			
t)	1:						
	11						
	2:						
	2:						
	2! 21						
	21						
	2!						
1 <sup>14</sup> 1 🔲							
rapp				Add Assertio	n		
ing/Courses/SELP/Songn L/tmp/uk.ac.ed.inf.son	ambı gnar				tmp/uk.ac.ed.inf.	songnambulist	
.songnambulist/uk.ac.e	d.i				oid.intent.catego	ry.LAUNCHER -D	
e: uk.ac.ed.inf.songna e: uk.ac.ed.inf.songna	mbu' mbu'						
ist	nsport	?		Cancel OK			

## The code generated by the test recorder (1/2)

@RunWith(AndroidJUnit4.class)
public class MapsActivityTest {

#### @Rule

public ActivityTestRule<MapsActivity> mActivityTestRule = new ActivityTestRule<>(MapsActivity.class);

#### @Test

```
public void mapsActivityTest() {
```

// Added a sleep statement to match the app's execution delay.
try { Thread.sleep(5000); }
catch (InterruptedException e) { ... }

#### // First button click

ViewInteraction floatingActionButton = onView( allOf(withId(R.id.fab), isDisplayed())); floatingActionButton.perform(click());

## The code generated by the test recorder (2/2)

#### // Second button click

}

// Added a sleep statement to match the app's execution delay.
try { Thread.sleep(5000); }
catch (InterruptedException e) { ... }

// Clicked on the overflow menu (:) in the app bar openActionBarOverflowOrOptionsMenu( getInstrumentation().getTargetContext());

## Adding unit tests

 We can then edit this test to add in JUnit assertions of the form assertTrue, assertFalse, assertNotNull, assertEquals, assertArrayEquals, and others.

import android.location.Location; import static org.junit.Assert.\*;

Location loc = mActivityTestRule.getActivity().getLocation(); assertTrue("Location is not null", loc != null);

## Making our app testable [in MapActivity]

- We may need to add some methods to our Activity to make values visible for testing.
- We can annotate these to show that they are used for testing. The annotation @VisibleForTesting prevents us from actually calling this method from production code.

import android.support.annotation.VisibleForTesting;

```
...
private Location mLastLocation;
...
@VisibleForTesting
public Location getLocation() {
return mLastLocation;
}
```

**Running tests** 

### **Running tests**

- Classes which contain tests are annotated with the annotation @RunWith(AndroidJUnit4.class) which means that they will be executed under the supervision of a test runner.
- This specifies the AndroidJUnitRunner class provided in the Android Testing Support Library as the default test runner.
- The emulator will start up as usual, but it will receive input events (such as button clicks) from our @Test methods.

## Running tests ....

_		ic notifications black 24dp.xml	31 // The recommended way to handle such scenarios is to use Espresso idling red
		ic sync black 24dn xml	32 // https://google.github.io/android-testing-support-library/docs/espresso/id
		T lavout	33 try {
		activity mans yml	34 Thread. sleep(5000);
		activity_maps.xm	35 ) catch (InterruptedException e) {
		menu	36 e.printstack(race();
		mipmap	37 J
	Run	MapsActivityTest	
	¢.	😑 🧮 🦊 🚝 포 🐳 🛧 🕂 🔃 🖳 » 😋	)
		Running tests	Testing started at 05:47
		v Quk.ac.ed.inf.songnambulist.MapsActi	
		mapsActivityTest	11/01 05:4/:44: Launching MapsActivitylest
		· · · · · · · · · · · · · · · · · · ·	We app changes detected since tast installation, skipping installation or (lisers/sto/Documents/Teachion/Courses/SFI P/Sononambulist/Sononambulist/anp/build/outputs/ank/anp-debup.ank
	`a.		S adb shell am force-stop w.ac.ed inf.sonambulist
	×		No apk changes detected since last installation, skipping installation of
	2		/Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/app-debug-androidTes
	1		§ adb shell am force-stop uk.ac.ed.inf.songnambulist.test
			Running tests
			s adb shell am instrument -w -r -e debug false -e class uk.ac.ed.inf.sonnnambulist.MansActivityTest uk.ac.ed.inf
			AndroidJUnitRunner
			Client not ready yet
		5	Started running tests
2			
ia.			
۲ą.			
P			
Bu			
۰			
ites			
vo			
P			
2			
*			
	▶.4	4: Run 🐞 5: Debug 😚 TODO 🛛 🗍 6: Android Moni	itor 🗵 Terminal 🧱 Q: Messages

## Running tests ... [Success]

		ic notifications black 24dp.xml	31 // The recommended way to handle such scenarios is to use Espresso idling red
		ic sync black 24dp.xml	32 // https://google.github.io/android-testing-support-library/docs/espresso/idl
	T	layout	33 try (
		activity maps.xml	34 Inread.sleep(Sw00);
	Þ	menu	35 etch (interruptedcepted e) (
	⊳	mipmap	37 }
	-0		38
R	un 🔛 Ma	apsActivityTest	
	• 😐	■ 禄 辰 至 美 全 🕂 🗹 🖳 🦇	1 test passed - 13s 11ms
	<b>v</b> (	Test Results 13s 11ms	Testing started at 05:47
		www.ac.ed.inf.songnambulist.MapsAct	11/01 05:47:44: Lounching MoneActivityTect
	Ξ.	mapsActivityTest	No and changes detected since last installation, skinning installation of
	2		/Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/app-debug.apk
			\$ adb shell am force-stop uk.ac.ed.inf.songnambulist
1	•		No apk changes detected since last installation, skipping installation of
	2		s adb shell am force-stop uk.ac.ed.inf.songnambulist.test
			Running tests
			s add shell am instrument -v -r -e debug false -e class uk.ac.ed.inf.songnambulist.MapsActivityTest uk.ac.ed.inf .AddreidJuminer Client not ready yet Started running tests Tests ran to completion.
🔹 Build Variants			
2: Favorites	▶ <u>4</u> : Run	🏶 Ş: Debug 👒 TODO 🍦 <u>6</u> : Android Mo	nitor 🔟 Terminal 🖉 🖻 Messages

## Running tests ....

_		ic notifications black 24dp.xml	31 // The recommended way to handle such scenarios is to use Espresso i
		ic_sync_black_24dp.xml	32 // https://google.github.io/android-testing-support-library/docs/esp
		v 🗖 layout	33 try {
		activity_maps.xml	34 Inread.step(SOVD); 35 } catch (InteruntedExcention e) {
		menu	36 e.printStackTrace();
		🕨 🗈 mipmap	37 }
	Run	MapsActivityTest	. 30
	Ċ	🐵 🗐 🐙 토 춘 🛧 🕈 🖬 🖳 » 🛛	
		Q Instantiating tests	Testing started at 05:46
			11/01 05:46:47: Launching MapsActivityTest
	62		<pre>\$ adb push /Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/ap \$ adb shell pm install -r "/data/local/tmp/uk.ac.ed.inf.songnambulist"</pre>
	Ç.		
	2		
	÷.		
ants			
l Var			
Build			
٠			
tes			
avori			
2.5			
*			
	<b>0</b> , 4	4: Run 🛛 🎄 5: Debug 🖓 TODO 🛛 🏺 <u>6</u> : Android Mor	nitor 🔟 Terminal 🧱 🛛: Messages

## Running tests ....

_		ic notifications black 24dn xml	21 // The recommended way to handle such scenarios is to use Espresse i
		is sync black 24dp yml	32 // https://aogale.aitub.io/android-testing-support-library/docs/esp
		= Dayout	33 try {
		antivity mana uml	34 Thread.sleep(5000);
		activity_maps.xmi	<pre>35 } catch (InterruptedException e) {</pre>
		menu	e.printstackirace();
		mipmap	38
	Run	KapsActivityTest	
	Ċ	🐵 🧮 🐙 토 조 중 🛧 🗜 🖸 🗟 »	)
		Instantiating tests	Testing started at 05:46
	-		11/01 05:46:47: Launching MapsActivityTest
	3		<pre>\$ adb push /Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/ap \$ adb shell pm install -r "/data/local/tmp/uk.ac.ed.inf.songnambulist"</pre>
	×		Success
	2		
	· ·		S adb push /Users/stg/Documents/leaching/Courses/SELP/Songnambulist/Songnambulist/app/bulld/outputs/apk/ap songnambulist test
			S adb shell pm install -r "/data/local/tmp/uk.ac.ed.inf.songnambulist.test"
2			
ant			
Var			
핕			
a l			
•			
S			
orit			
Fav			
Ň			
*			
	▶, 4	🛓: Run 🛛 🎄 💈: Debug 🛛 🛬 TODO 🛛 🌞 🙆: Android Mor	nitor 🔟 Terminal 🧮 🖞: Messages

## Running tests ...

_		ic_notifications_black_24dp.xml	31 // The recommended way to handle such scenarios is to use Espresso i
		ic_sync_black_24dp.xml	32 // https://google.github.io/android-testing-support-library/docs/esp
		🔻 🛅 layout	33 try {
		activity_maps.xml	35 } cate(interprintedException e) {
		menu	<pre>36 e.printStackTrace();</pre>
		🕨 🛅 mipmap	37 }
	Run	MapsActivityTest	38
	Ċ	🐵 🗐 ↓2 県 王 축 🛧 🖡 🗹 🖳 » (	
		▼ 🥥 Running tests	Testing started at 05:46
		<ul> <li>Uk.ac.ed.inf.songnambulist.MapsActi</li> <li>mapsActivityTest</li> </ul>	11/01 05:46:47: Launching MapsActivityTest \$ adb push /Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/ap
	je <sup>n</sup>		<pre>\$ adb shell pm install -r "/data/local/tmp/uk.ac.ed.inf.songnambulist"</pre>
	×		SUCCESS
	?		S adb push /Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/ap
			.songnambulist.test \$ adb shell pm install -r "/data/local/tmp/uk.ac.ed.inf.songnambulist.test" Success
			Running tests
			S adb shell am instrument → -r -e debug false -e class uk.ac.ed.inf.songnambulist.MapsActivityTest uk.a .NafroidJNitRunner Client not ready yet. Sarred running tests
s			
lant			
N N			
Sulf			
4			
\$			
orite			
Fave			
Ň			
*			
	1,4	4: Run 🛛 🐞 5: Debug 🖓 TODO 🛛 🏺 6: Android Mon	itor 📧 Terminal 🗮 Q: Messages

## Running tests ... [Failure]

		ic notifications black 24dp.xml	31 // The recommended way to handle such scenarios is to use Espresso i
		ic sync black 24dp.xml	32 // https://google.github.io/android-testing-support-library/docs/esp
		= Diavout	33 try {
		• El layout	34 Thread.sleep(5000);
		activity_maps.xmi	35 } catch (InterruptedException e) {
		menu	<pre>36 e.printStackTrace();</pre>
		mipmap	3/ <b>J</b> 38
	Run	MapsActivityTest	
		🐵 🧮 ↓2 ঢ় 조 축 ↑ 🕇 🖸 🖳 »	1 test failed - 14s 301ms
		v O Terminated 14s 301ms	Testing started at 05:46
		v output outp	
		mapsActivityTest 14s 301ms	11/01 05:46:47: Launching MapsActivityTest
			5 adb push / users/sty/bc/uments/reaching/courses/sci/songnamoutis//songnamoutis//app/outu/outputs/apk/ap
	ìa.		S ado snect più Instatt - 1 / data/ totat/ tiip/ uk/at/ed. Int. Songhanud tist
	×		
	2		
	1		\$ adb push /Users/stg/Documents/Teaching/Courses/SELP/Songnambulist/Songnambulist/app/build/outputs/apk/ap
			.songnambulist.test
			S add shell pm install -r "/data/local/tmp/uk.ac.ed.int.songnamoulist.test"
			Juccess
			Running tests
			\$ add shell an instrument -w -r -e debug false -e class uk.ac.ed.inf.songnambulist.MapsActivityTest uk.a Android Ukielenergy
			Android Unitkunner
			Started running tests
			started running costs
			Test failed to run to completion. Reason: 'Instrumentation run failed due to 'Process crashed.''. Check de
			Test running failed: Instrumentation run failed due to 'Process crashed.'
3			
lan			
Na l			
Pi			
ã			
~			
ite			
2			
Ë.			
7			
1			
	1	🛓: Run 🛛 🐞 5: Debug 🖓 TODO 🛛 🌞 6: Android Mor	nitor 🔟 Terminal 💻 Q: Messages

### When tests fail

- Tests (especially tests generated by the Espresso test recorder) can fail for reasons other than an error in our application logic. It is important to look at the reason why the test failed; it might be a poorly-specified test.
- False positive failures can be caused by timing issues where the app under test does not respond within the delay anticipated by the sleep pause in the test.
- Work is underway to improve the Android testing framework.

## Links

- https://developer.android.com/training/testing/index.html
- https://developer.android.com/training/testing/fundamentals.html
- https://developer.android.com/training/testing/junit-rules.html
- https://developer.android.com/training/testing/espresso/index.html
- https://developer.android.com/studio/write/annotations.html