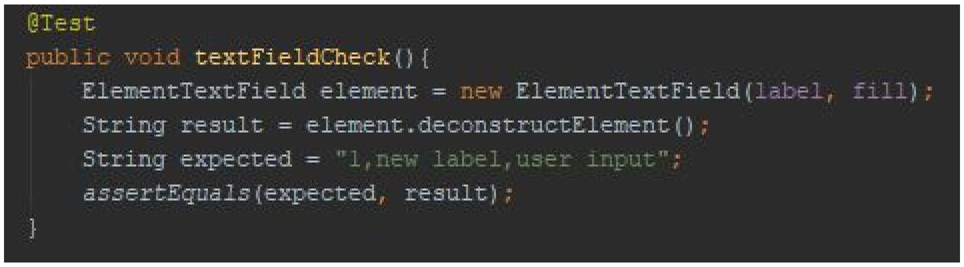
Binary Giant

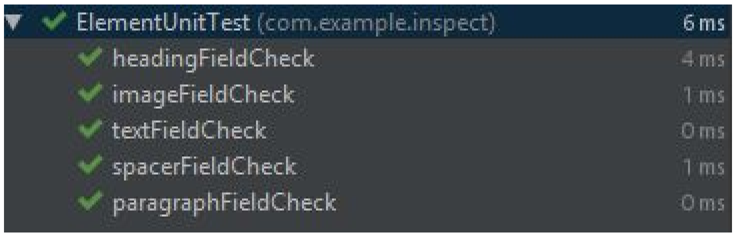
Inspect Testing Report

Unit tests

Traditional unit tests can only be performed on pure java classes. As such for these we could only test our data objects that hold UI data across the UI lifecycle. Each element was tested to ensure that data was being returned in the correct format required for saving and loading. Due to the nature of these objects simply being containers for Strings the tests are quite straight forward.



```
@Test
public void textFieldCheck(){
    ElementTextField element = new ElementTextField(label, fill);
    String result = element.deconstructElement();
    String expected = "1,new label,user input";
    assertEquals(expected, result);
}
```

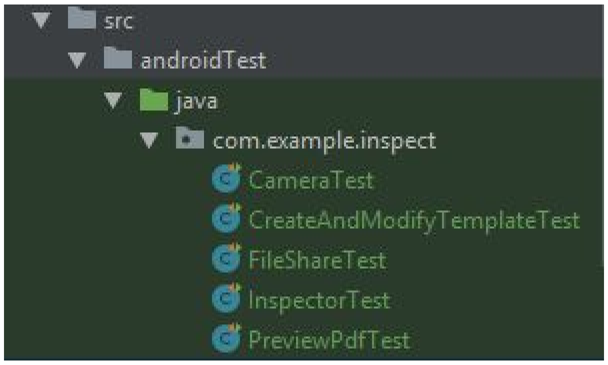


```
▼ ✓ ElementUnitTest (com.example.inspect)    6 ms
      ✓ headingFieldCheck                     4 ms
      ✓ imageFieldCheck                       1 ms
      ✓ textFieldCheck                        0 ms
      ✓ spacerFieldCheck                      1 ms
      ✓ paragraphFieldCheck                   0 ms
```

There are needlessly complicated ways to do unit tests within an android app however all our data is shown in the UI and our integration tests can easily assert and validate the data in the UI making it a little redundant for the amount of effort considering our testing environment.

Instrument (Integration) Tests

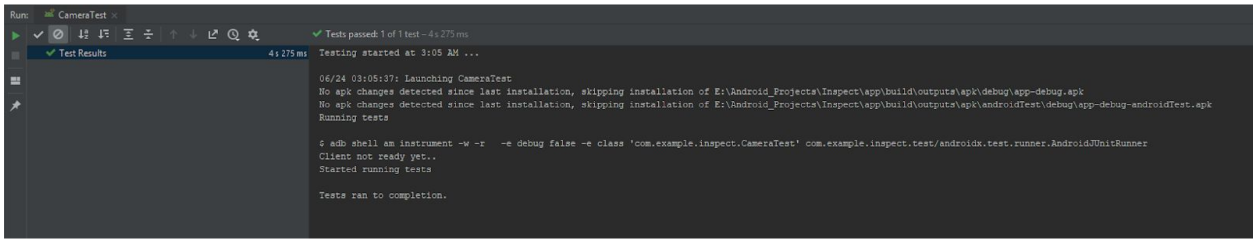
Almost all our classes extend AppCompatActivity meaning that they are all what android considers an “Activity” with a UI composed of views. All activities are launched with an intent. What this means is that all activities are very loosely coupled and the only integration that can occur is launching other activities. For this we used Espresso to simulate user interactions to run through the activities ensuring the expected activities were launched with the correct UI views displayed. Assertions are made to validate data and confirm data binding is occurring with the pure java data objects.



```
▼ src
  ▼ androidTest
    ▼ java
      ▼ com.example.inspect
          CameraTest
          CreateAndModifyTemplateTest
          FileShareTest
          InspectorTest
          PreviewPdfTest
```

Unfortunately external activities are impossible to track (androids camera for example), but luckily they are outside the scope of our testing.
Another downside is these tests are unusually lite on details when they pass.

CameraTest shows that the PhotoManager activity can be launched from the camera button element, and that the android camera can be accessed.



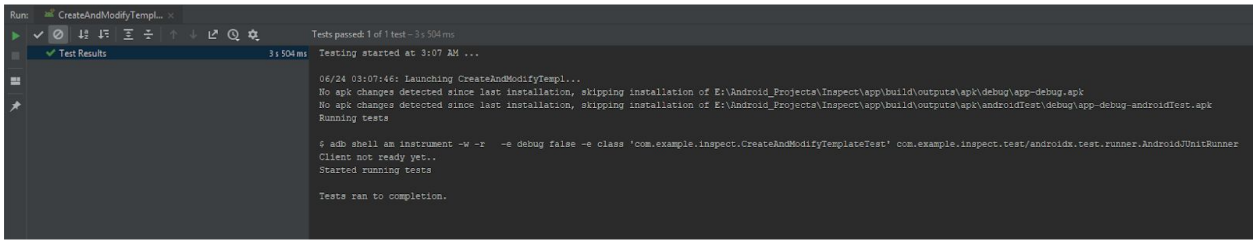
```
Testing started at 3:05 AM ...

06/24 03:05:37: Launching CameraTest
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\debug\app-debug.apk
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\androidTest\debug\app-debug-androidTest.apk
Running tests

$ adb shell am instrument -w -r   -e debug false -e class 'com.example.inspect.CameraTest' com.example.inspect.test/androidx.test.runner.AndroidJUnitRunner
Client not ready yet..
Started running tests

Tests ran to completion.
```

CreateSaveModifyTemplateTest creates a template then modifies and saves it. Filemanager is then launched and the template is reloaded and modified again. The formatting is compared to ensure the template is loaded correctly.



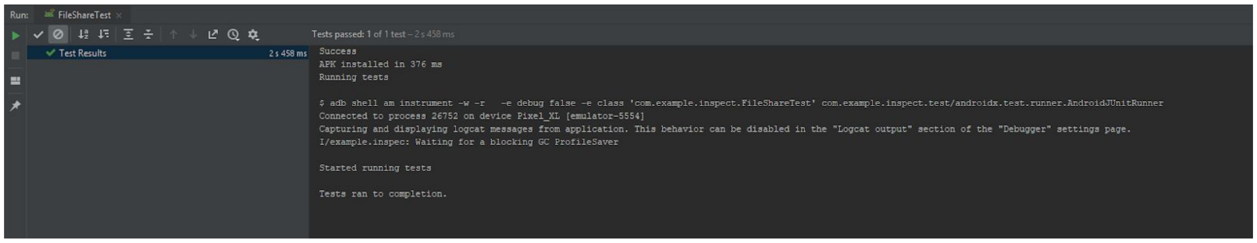
```
Testing started at 3:07 AM ...

06/24 03:07:46: Launching CreateAndModifyTempl...
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\debug\app-debug.apk
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\androidTest\debug\app-debug-androidTest.apk
Running tests

$ adb shell am instrument -w -r   -e debug false -e class 'com.example.inspect.CreateAndModifyTemplateTest' com.example.inspect.test/androidx.test.runner.AndroidJUnitRunner
Client not ready yet..
Started running tests

Tests ran to completion.
```

An external file sharing activity is launched from the FileManager activity. It is understandably a straight forward test.



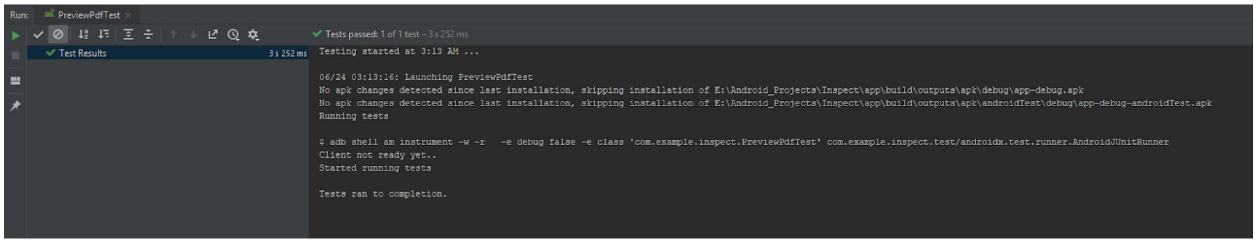
```
Success
APK installed in 376 ms
Running tests

$ adb shell am instrument -w -r   -e debug false -e class 'com.example.inspect.FileShareTest' com.example.inspect.test/androidx.test.runner.AndroidJUnitRunner
Connected to process 26752 on device Pixel_XL [emulator-5554]
Capturing and displaying logcat messages from application. This behavior can be disabled in the "Logcat output" section of the "Debugger" settings page.
I/example.inspec: Waiting for a blocking GC ProfileSaver

Started running tests

Tests ran to completion.
```

The ViewPrintAdapter launches an external print to pdf activity. This is done from Inspect in editing mode.



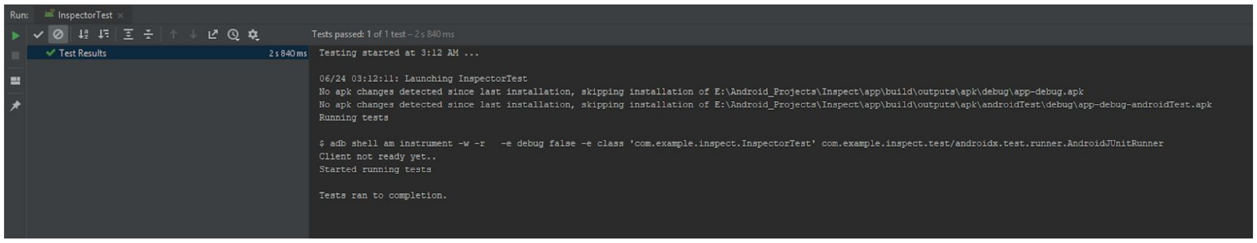
```
Testing started at 3:13 AM ...

06/24 03:13:16: Launching PreviewPdfTest
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\debug\app-debug.apk
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\androidTest\debug\app-debug-androidTest.apk
Running tests

$ adb shell am instrument -w -r   -e debug false -e class 'com.example.inspect.PreviewPdfTest' com.example.inspect.test/androidx.test.runner.AndroidJUnitRunner
Client not ready yet..
Started running tests

Tests ran to completion.
```

This test is FileManager opening a template in Inspect and Inspect accepting user input from the android keyboard ensuring data binding is occurring during inspection mode and can be saved to the template.



```
Testing started at 3:12 AM ...

06/24 03:12:11: Launching InspectorTest
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\debug\app-debug.apk
No apk changes detected since last installation, skipping installation of E:\Android_Projects\Inspect\app\build\outputs\apk\androidTest\debug\app-debug-androidTest.apk
Running tests

$ adb shell am instrument -w -r   -e debug false -e class 'com.example.inspect.InspectorTest' com.example.inspect.test/androidx.test.runner.AndroidJUnitRunner
Client not ready yet..
Started running tests

Tests ran to completion.
```