

Software Engineering Lab

Software Testing Specification

# **Course Assistant for Educators**

---

Streamlining the teaching process

Dibyadarshan Hota (16CO154)

Omkar Prabhu (16CO233)

April 14, 2018

## 1. Introduction

This document is the Software Test Specification for the Course Assistant for Educators. It will focus on the testing majority of the features implemented based on the requirements mentioned in the SRS document.

The testing plan includes Manual testing, Automated Testing and a combination of both for the features listed.

Travis CI used while developing ensures that code is correctly building at any point of time.

Results for the testing done have been provided.

## 2. Features to be Tested

ID	Features	Description
1	User registration	A user should be able to register through the mobile application. The user should provide password and a valid e-mail address. Internet connectivity will be required for registering.
2	Add course	A user should be able to create a new course providing the course details which should include: 1) No of students 2) Course code 3) Course Name 4) Contact of CR/TA
3	Record Attendance	A user should now be able to add attendance records for that course for the current course.
4	Add Marks	A user should now be able to add marks of students for that course. It should have entry for in-semester and end-semester marks

5	Add Project Deadlines	The user should now be able to add project deadlines for that course. The user should be able to add date and time for submission of the project along with a period of time before which the user wants to receive the notification
6	Display marks and attendance percentage	View all course data along with the student as a list containing student , marks and attendance percentage.
7	Search by student in list of students	User should be able to search by a students name or roll no in the students view of listing marks and attendance percentage along a student.
8	Sort in list of students	User should be able to sort in the students view of listing marks and attendance percentage along a student by: 1) Marks obtained. 2) Attendance percentage
9	View Attendance for a day	A user should be able to pick a date and view the attendance for that day
10	Contact CR / TA	Given that a user has added a course and contact information of CR/TA, the user should be able to contact them
11	Document Similarity	Allow the user to take two documents as input and display the magnitude of similarity between them
12	Feedback	A user should now be able to send feedback regarding the application. Internet connectivity should be present.

### 3. Approach

#### Overall approach to testing

Testing would follow two procedures

1) Automated Testing using Firebase Test Lab for Android service (Robo Test):

A Robo script is feeded into the Test Lab along the apk file of the application to be tested. After the test we get access to

- a) Issues
- b) Logs
- c) Screenshots of Testing
- d) Activity Map (Flow of Android Activities on various events performed by the Robo Test)
- e) Entire test recorded and video is provided
- f) Performance of the application: CPU usage, Network stats and Memory usage graphs are provided

2) Manual Testing

Feature like checking database entries and the corner cases for various form fields

Both of these methods will focus on covering all aspects of the features which includes:

- 1) UI functioning i.e. whether each event or action results into the desirable result. For example: On clicking of Add new Course Floating Action Button results in the screen which is Add Course Fragment.
- 2) Tasks corresponding to the those UI elements are working correctly. For example: On clicking of Save, the course is actually present as the table entry in the database.



## Approach for each feature

### 1) User registration

**Testing Approach:** Manual Testing

For various combinations of valid and invalid inputs during the authentication procedure Checking creation of user in Firebase Authentication.

### 2) Add course

**Testing Approach:** Robo Test for UI and Manual Testing for checking database entry

UI functionality will be checked by Robo Test and addition of course into the database will have to checked manually logging in ADB

### 3) Record Attendance

**Testing Approach:** Robo Test for UI and Manual Testing for checking database entry

UI functionality will be checked by Robo Test and addition of attendance into the database will have to checked manually logging in ADB

### 4) Add Marks

**Testing Approach:** Robo Test for UI and Manual Testing for checking database entry

UI functionality will be checked by Robo Test and addition of marks into the database will have to checked manually logging in ADB

### 5) Add Project Deadlines

**Testing Approach:** Manual Testing

Project Deadlines uses a external Calendar application to store deadlines and reminders and using Robo Test it is possible to test features of only the application provided. So manual testing will be done for these and checking whether the deadlines and reminders are set.



## 6) Display marks and attendance percentage

**Testing Approach:** Robo Test

UI functionality will be checked by Robo Test whether all marks and attendance are displayed.

## 7) Search by student in list of students

**Testing Approach:** Robo Test

UI functionality will be checked by Robo Test by providing default values for form fields in the Robo Script.

## 8) Sort in list of students

**Testing Approach:** Robo Test

UI functionality will be checked by Robo Test to check whether students are sorted by

- 1) Attendance percentage
- 2) End semester marks

## 9) View Attendance for a day

**Testing Approach:** Robo Test

Robo Test will check the UI functionality and check correct results display for the attendance for the chosen day

## 10) Contact CR / TA

**Testing Approach:** Manual Testing

As Robo Test does not work for external applications like mail, we have to test them manually whether mail template from application works correctly.

## 11) Document Similarity

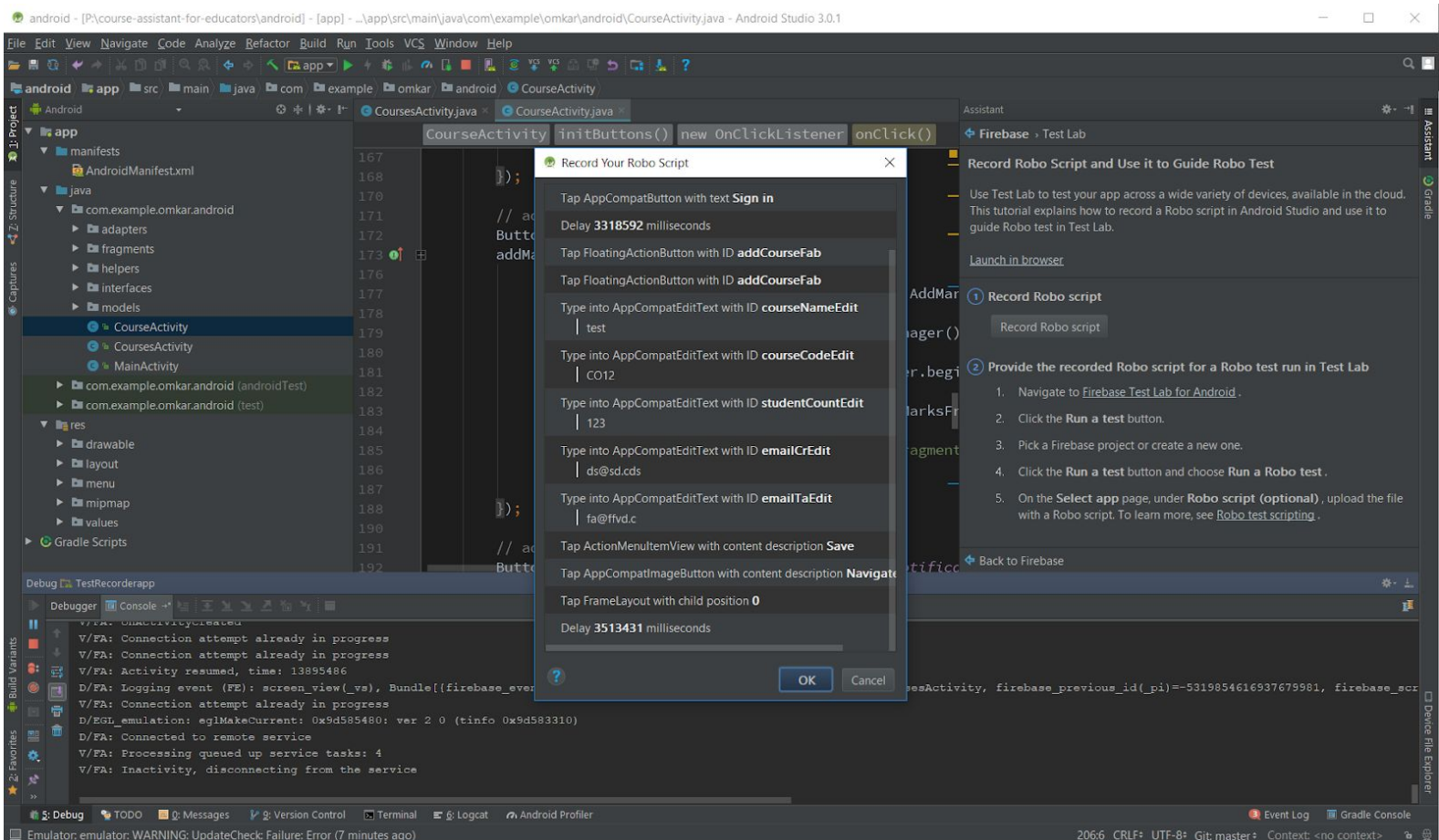
**Testing Approach:** Robo Testing

Providing default values for those input fields in the Robo Script we can test document similarity result is displayed or not.

## 12) Feedback

### Testing Approach: Robo Testing

After providing default values through the script to form input for feedback and suggestions we can test the functionality working.



## 4. Testing Results

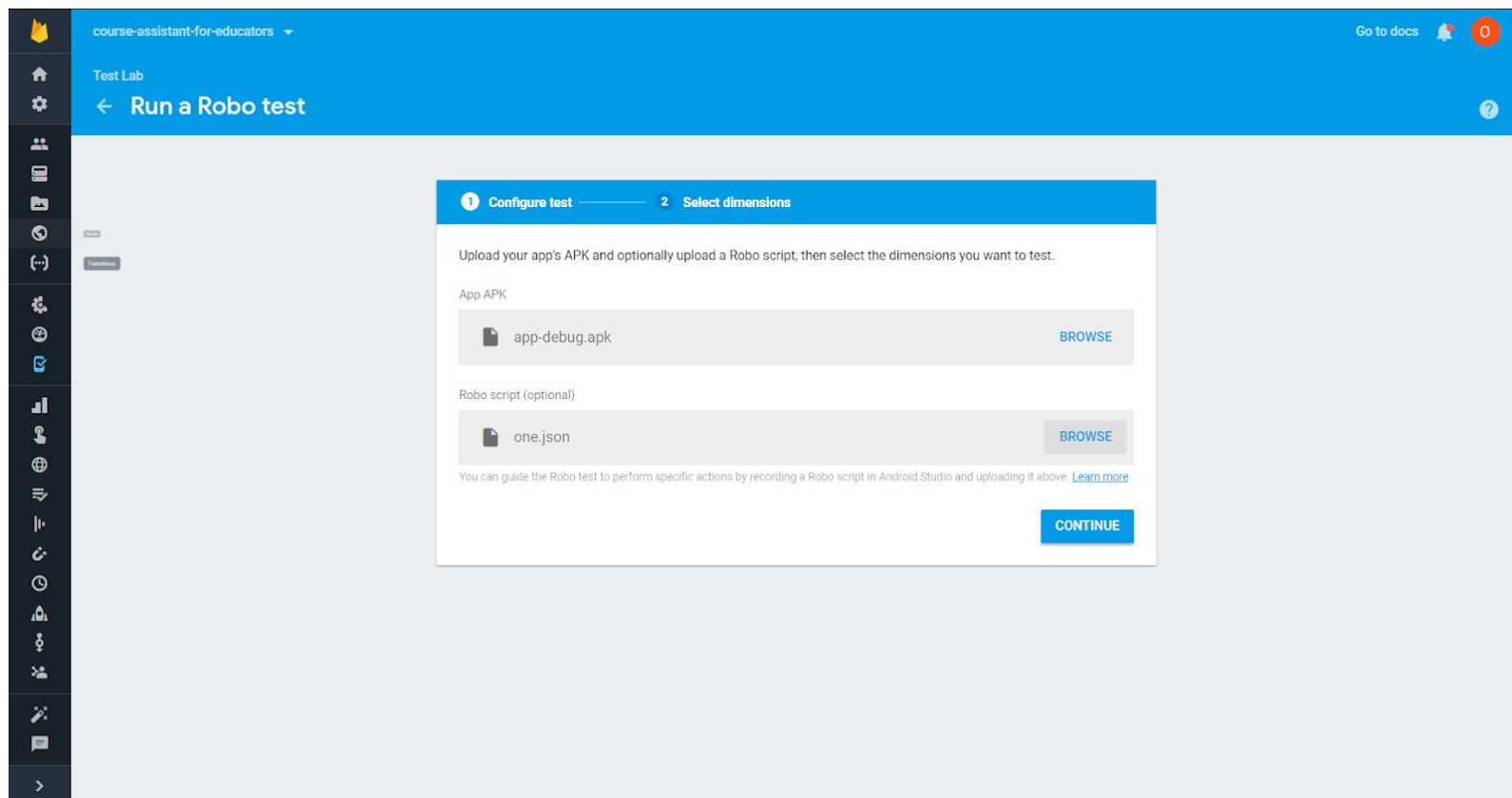
- Manual Testing for all forms and database entries and queries have been checked and integrated into the code here: [Course Assistant for Educators](#).
- Along with continuous integration from Travis CI we ensure that code builds correctly at any given time during the development stage.

The screenshot displays the Travis CI interface for the repository **CSE-Projects / course-assistant-for-educators**. The build status is **passing** for build #47. The build log shows the following steps:

```
1 Worker information
6 Build system information
528
529 Network availability confirmed.
530
531 127.0.0.1 localhost nettuno travis vagrant
532 127.0.1.1 travis-job-605d6eae-1cbe-46ea-83f2-0e9010c483af travis-job-605d6eae-1cbe-46ea-83f2-0e9010c483af ip4-loopback trusty64
533
534 $ jdk_switcher use oraclejdk8
535 Switching to Oracle JDK8 (java-8-oracle). JAVA_HOME will be set to /usr/lib/jvm/java-8-oracle
536 $ git clone --depth=50 --branch=master https://github.com/CSE-Projects/course-assistant-for-educators.git CSE-
546
547 Setting environment variables from .travis.yml
```



- Automated Testing using Firebase Test Lab for Android service (Robo Test) :
- 1) Initial Setup



course-assistant-for-educators

Test Lab > Run a Robo test

Go to docs

0

Physical devices

Nexus 7 (2013) ASUS

☐ 19 ☐ 21

Pixel 2 Google

☐ 26 ☒ 27

Nexus 9 HTC

☐ 21

Huawei P8 lite HUAWEI

☐ 21

Nexus 5 LG

☒ 19 ☐ 21 ☐ 22 ☐ 23

LG G6 LGUS997 LGE

☒ 24

Moto G Play (4th Gen) XT1607 Motorola

☐ 23

Moto G4 Plus Motorola

☐ 23

Nexus 6 Motorola

☐ 21 ☐ 22 ☐ 23

SH-04H SHARP

☐ 23

Galaxy J5 Samsung

☐ 23

Pixel Google

☐ 25 ☐ 26

HTC One (M8) HTC

☐ 19

Huawei Mate 9 HUAWEI

☐ 24

LG G3 LG

☐ 19

LG K4 (LG-X230) LG Electronics

☐ 23

Moto G (3rd Gen) Motorola

☒ 22

Moto G4 Motorola

☒ 23

Moto X Motorola

☐ 19

OnePlus One OnePlus

☐ 22

Galaxy J1 ace SM-J111M Samsung

☐ 22

Galaxy J7 (SM-J710MN) Samsung

☐ 23

Dashboard

course-assistant-for-educators

Test Lab > android

Go to docs

0

matrix-11kw6zebbz1ab

Robo test, Just now

CANCEL

Failed

Passed

Skipped

Inconclusive

0

0

0

0

Test execution	Duration	Locale	Orientation	Issues
Moto G (3rd Gen), API Level 22	—	English (United States)	Portrait	—
Pixel 2, API Level 27	—	English (United States)	Portrait	—
Nexus 5, API Level 19	—	English (United States)	Portrait	—
LG G6 LGUS997, API Level 24	—	English (United States)	Portrait	—
Moto G4, API Level 23	—	English (United States)	Portrait	—

## 2) Running Matrix

course-assistant-for-educators

Test Lab for Android

TESTS

TEMPLATES

Go to docs

0

Want unlimited tests? [Upgrade](#)

android

RUN A TEST

Test matrix	Test type	Started	Total executions	Issues
matrix-11kw6zebbz1ab	Robo	Just now	5	—
matrix-1mzm78ac22f9y	Robo	7 days ago	1	—
matrix-1hedmg6sgrgl	Robo	7 days ago	1	—
matrix-jgb49wsjel6y	Robo	7 days ago	1	—
matrix-72dws1phkyme	Robo	7 days ago	2	—
matrix-3obhjrqqzufuv	Robo	9 days ago	1	—

### 3) Logs

course-assistant-for-educators

Test Lab > android > matrix-1mzm78ac22f9y

← Robo test, Pixel 2, API Level 27

Go to docs

0

Passed

4/6/18, 4:35 PM

4 min 5 sec

Portrait

English (United States)

VIEW SOURCE FILES

TEST ISSUES

LOGS

SCREENSHOTS

ACTIVITY MAP

VIDEOS

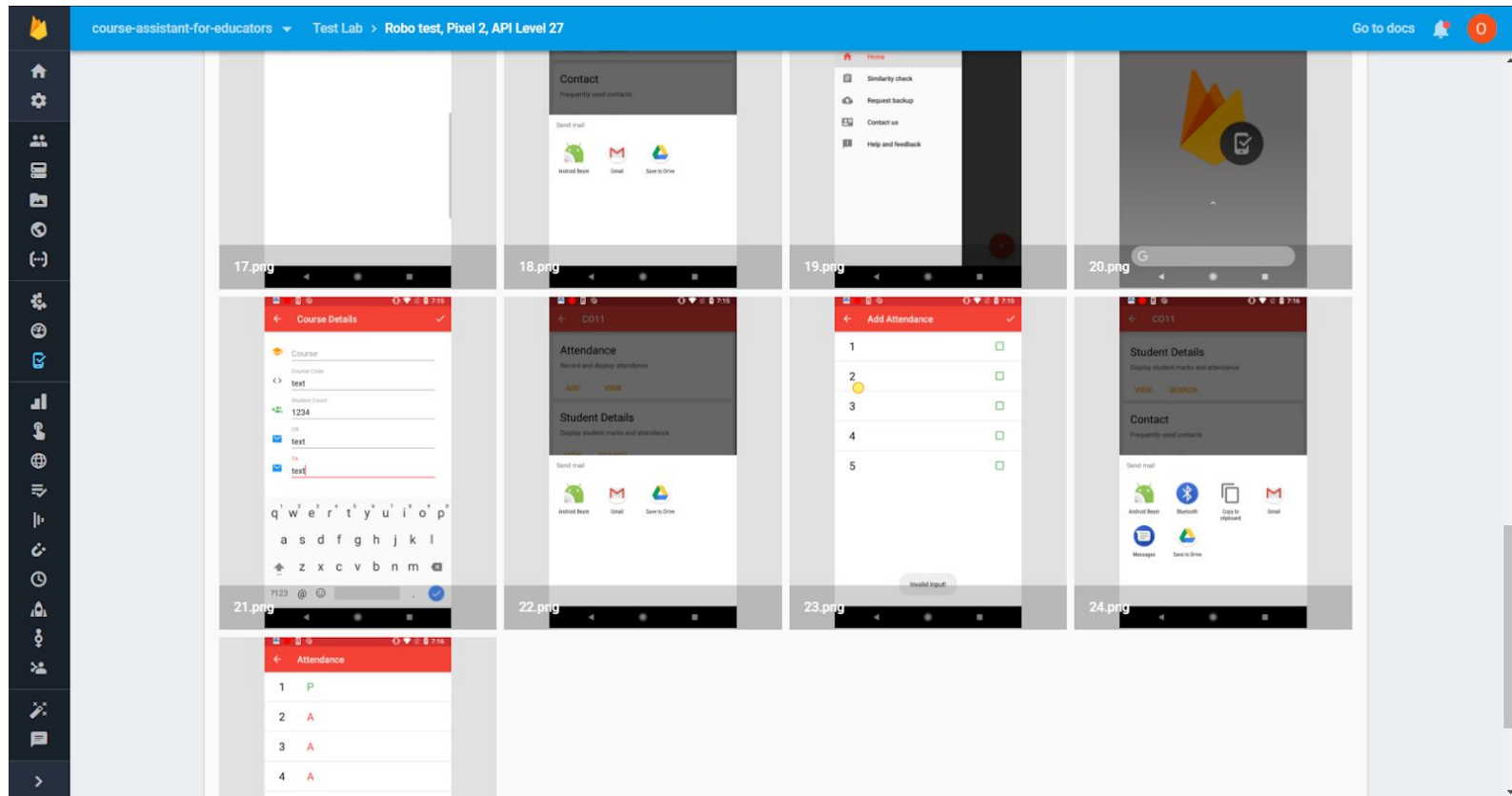
PERFORMANCE

Warning and higher

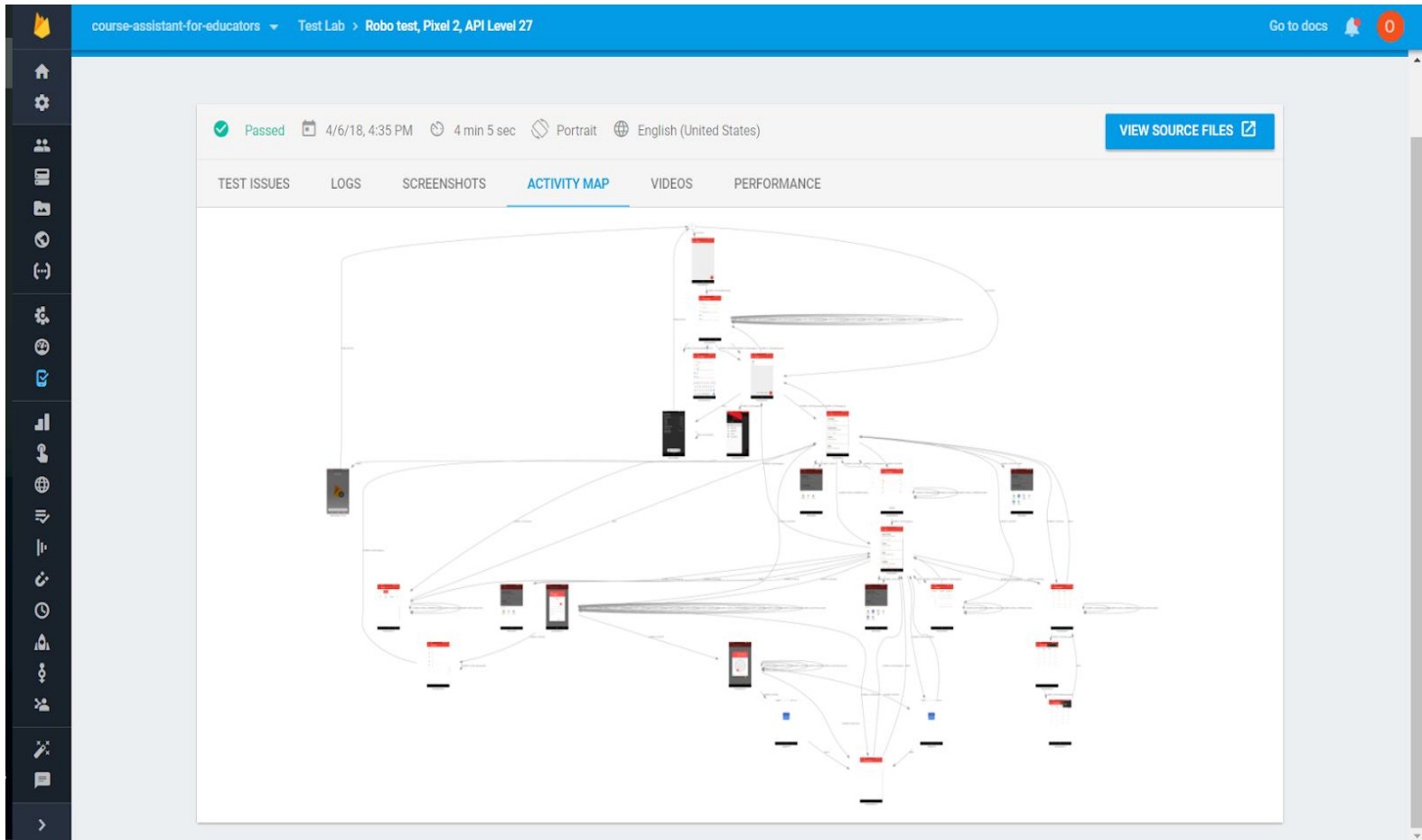
Styled

W	07:12:17.341	sh	type=1400 audit(0.0:141): avc: denied { read } for name="cache" dev="dm-0" ino=25 scontext=u:r:shell:s0 tcontext=u:object_r:cache_file:s0 tclass=lnk_file permissive=0
E	07:12:18.191	asset	setgid: Operation not permitted
E	07:12:19.112	installD	Failed to delete /data/app/vmdl699866463.tmp: No such file or directory
W	07:12:19.142	BroadcastQueue	Background execution not allowed: receiving Intent { act=android.intent.action.PACKAGE_ADDED dat=package:com.google.android.gms.policy_test_support flg=0x4000010 (has extras) } to com.android.musicfx/.Compatibility\$Receiver
E	07:12:19.170	VvmPkgInstalledRc...	cannot create TelephonyManager from ComponentInfo{com.android.phone/com.android.services.telephony.TelephonyConnectionService}, [****], UserHandle{0}
W	07:12:19.171	sh	type=1400 audit(0.0:142): avc: denied { read } for name="cache" dev="dm-0" ino=25 scontext=u:r:shell:s0 tcontext=u:object_r:cache_file:s0 tclass=lnk_file permissive=0
W	07:12:19.258	Finsky(13168): [2] ...	No account configured on this device.
W	07:12:19.369	...	No account configured on this device.
W	07:12:19.372	Finsky(13168): [65] ...	No account configured on this device.
W	07:12:19.379	...	No account configured on this device.
W	07:12:19.379	Finsky(13168): [2] ...	No account configured on this device.

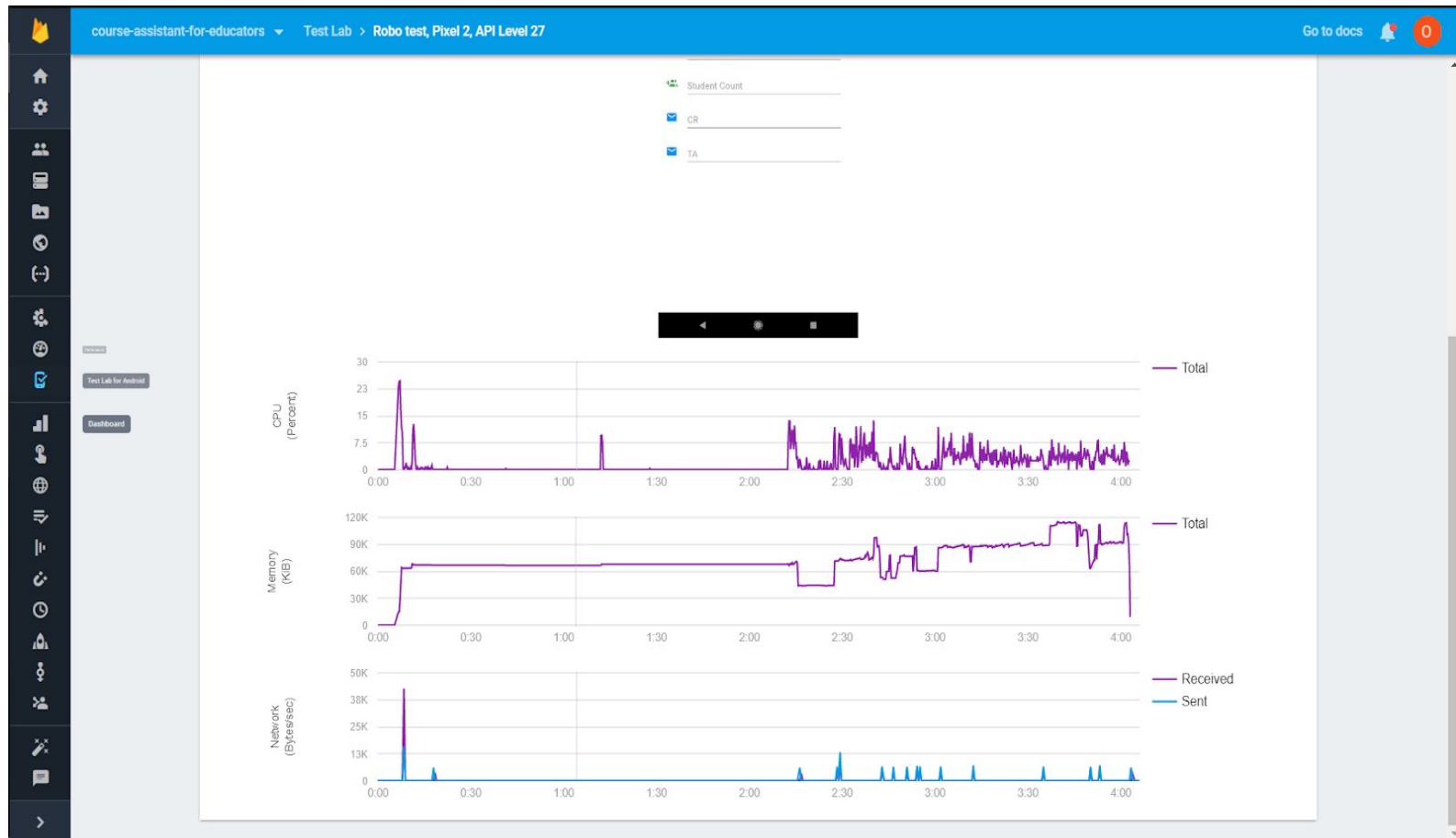
#### 4) Testing screenshots by Robo Test



## 5) Activity Diagram generated by Robo Test



## 6) Performance of the application: CPU usage, Network stats and Memory usage graphs



## 7) Entire test recorded and video is provided

✓ Passed

4/6/18, 4:35 PM

4 min 5 sec

Portrait

English (United States)

VIEW SOURCE FILES

TEST ISSUES

LOGS

SCREENSHOTS

ACTIVITY MAP

VIDEOS

PERFORMANCE

★ Videos now show you where Robo taps, to help you follow along

Course Details

Course

test

<> Course Code

+ Student Count

CR

TA