

REPORT

Codelab 5: Activity lifecycle and state

Subject: Desarrollo de Aplicaciones

Moviles Nativas

Group : 7CV1

Student : Julio Cesar Hernández Reyes

Teacher : Velez Saldaña Ulises



1. Introduction

In this practical i created explicit intents. Worked and edited Activities. And use buttons and text-views to send messages from one Activity to another

In this practical i learned more about the activity lifecycle. The lifecycle is the set of states an activity can be in during its entire lifetime, from when it's created to when it's destroyed and the system reclaims its resources.

As a user navigates between activities in an app (as well as into and out of the app), activities transition between different states in their lifecycles.

<https://developer.android.com/codelabs/android-training-activity-lifecycle-0>

What i learned:

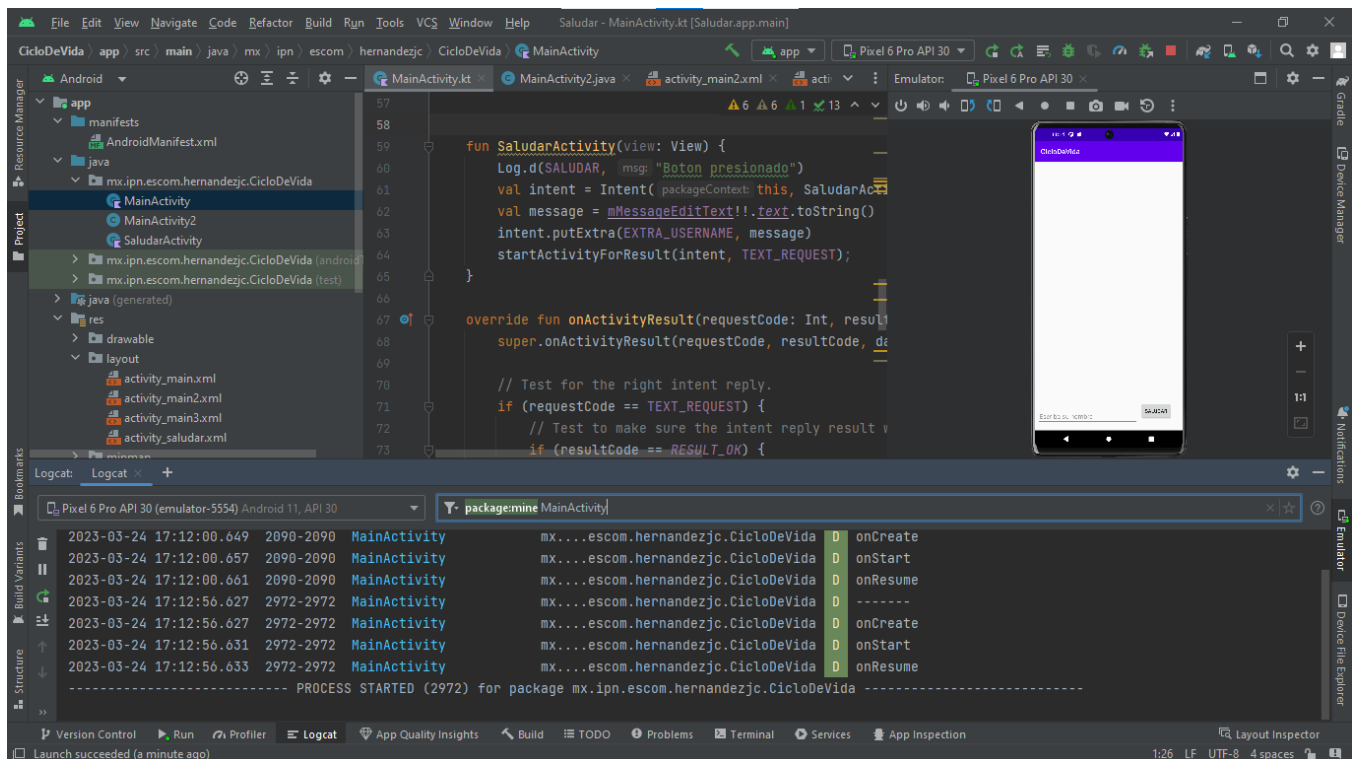
- How the Activity lifecycle works.
- When an Activity starts, pauses, stops, and is destroyed.
- About the lifecycle callback methods associated with Activity changes.
- The effect of actions (such as configuration changes) that can result in Activity lifecycle events.
- How to retain Activity state across lifecycle events.

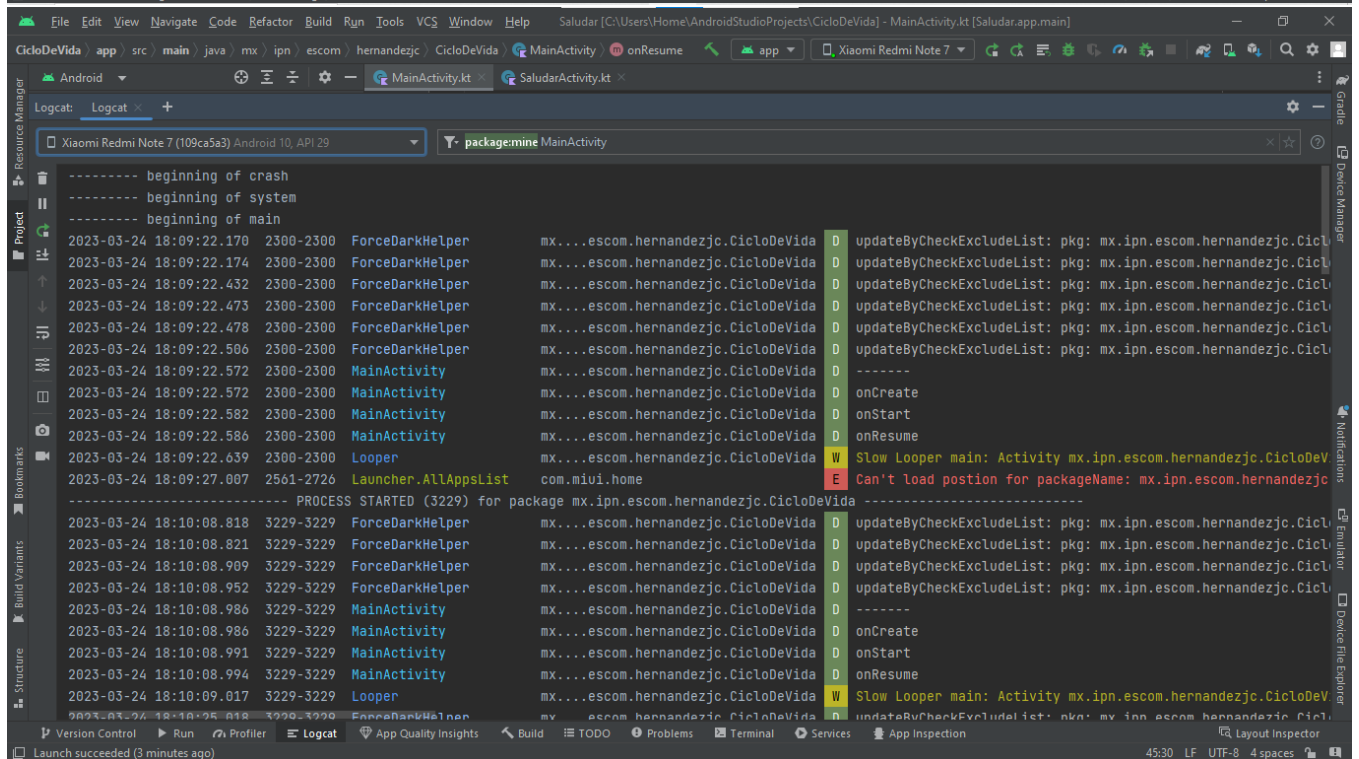
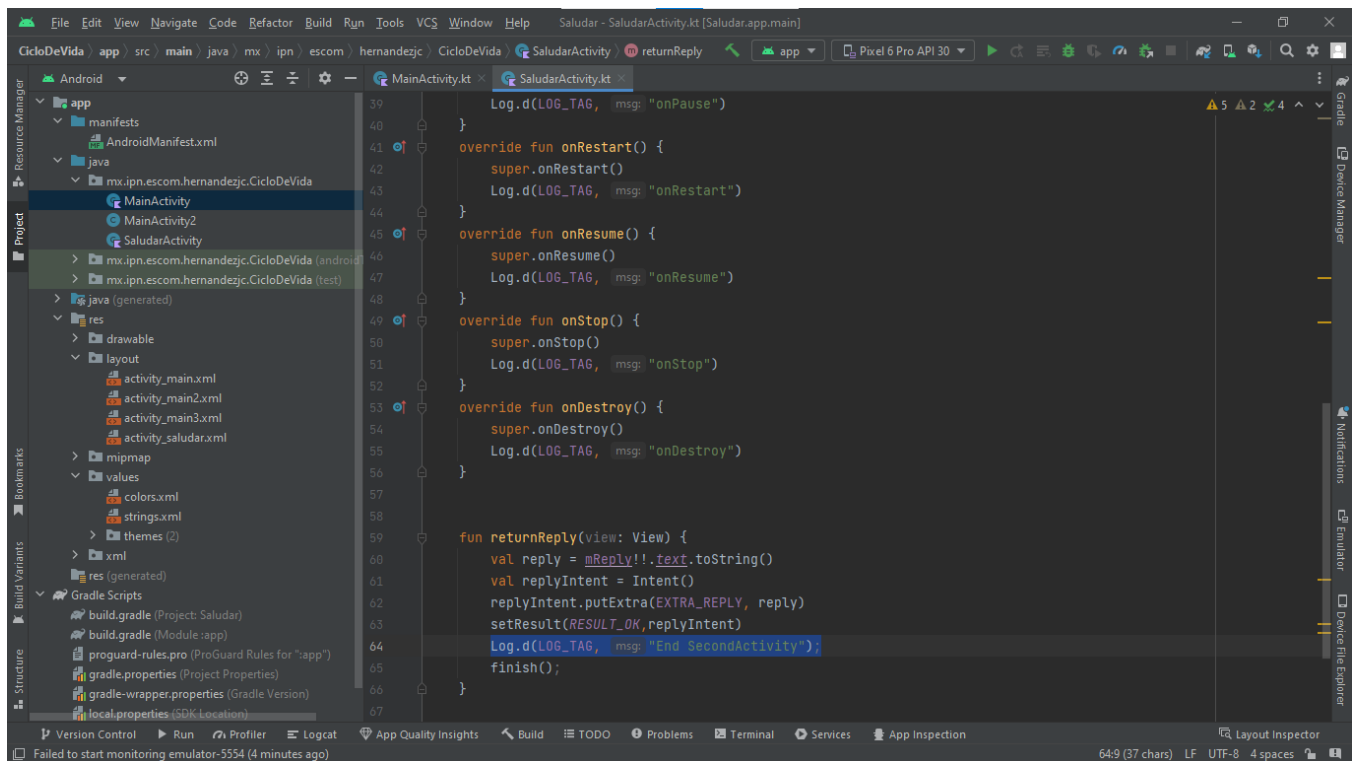
2. Development

2.1. Tasks

2.1.1. Task 1: Add lifecycle callbacks to TwoActivities

In this task i implemented all of the Activity lifecycle callback methods to print messages to logcat when those methods are invoked. These log messages allowed me to see when the Activity lifecycle changes state, and how those lifecycle state changes affected my app as it runs.



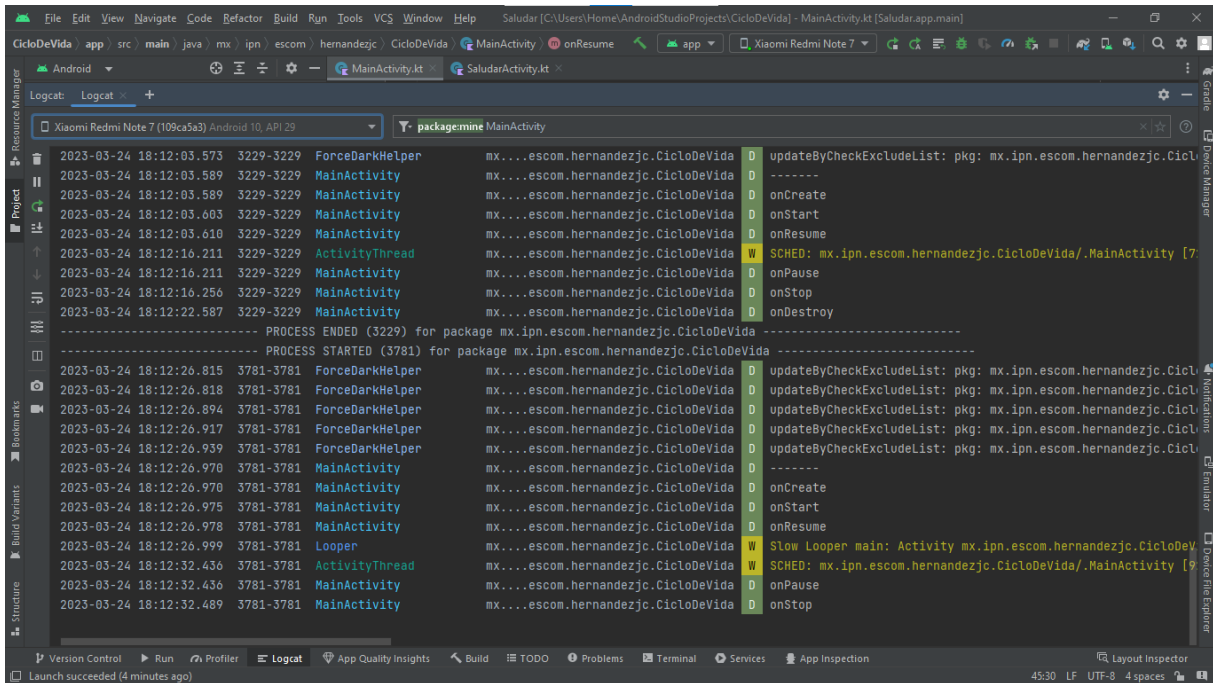


Android Studio interface showing the Logcat window. The top bar indicates the project is 'CicloDeVida' and the selected activity is 'MainActivity'. The Logcat window is filtered to show logs for 'package:mine.MainActivity' on a 'Xiaomi Redmi Note 7 (109ca5a3) Android 10, API 29'. The log entries show the lifecycle of the MainActivity, including onCreate, onStart, onResume, onPause, onStop, onDestroy, and onRestart. The log also shows the execution of the 'Slow Looper main' and the 'updateByCheckExcludeList' method. The log entries are as follows:

Time	Thread	Class	Message
2023-03-24 18:10:08.986	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onCreate
2023-03-24 18:10:08.991	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStart
2023-03-24 18:10:08.994	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onResume
2023-03-24 18:10:09.017	3229-3229	Looper	mx...escom.hernandezjc.CicloDeVida W Slow Looper main: Activity mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:10:25.018	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:10:28.955	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D Boton presionado
2023-03-24 18:10:28.997	3229-3229	ActivityThread	mx...escom.hernandezjc.CicloDeVida W SCHED: mx.ipn.escom.hernandezjc.CicloDeVida/.MainActivity [8]
2023-03-24 18:10:28.998	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onPause
2023-03-24 18:10:29.287	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStop
2023-03-24 18:10:43.362	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onRestart
2023-03-24 18:10:43.363	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStart
2023-03-24 18:10:43.365	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onResume
2023-03-24 18:10:47.859	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D Boton presionado
2023-03-24 18:10:47.880	3229-3229	ActivityThread	mx...escom.hernandezjc.CicloDeVida W SCHED: mx.ipn.escom.hernandezjc.CicloDeVida/.MainActivity [7]
2023-03-24 18:10:47.880	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onPause
2023-03-24 18:10:48.070	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStop
2023-03-24 18:10:52.435	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onDestroy
2023-03-24 18:10:52.466	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:10:52.495	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:10:52.503	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D -----
2023-03-24 18:10:52.503	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onCreate
2023-03-24 18:10:52.507	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStart
2023-03-24 18:10:52.510	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onResume
2023-03-24 18:11:12.539	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onPause
2023-03-24 18:11:12.541	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStop

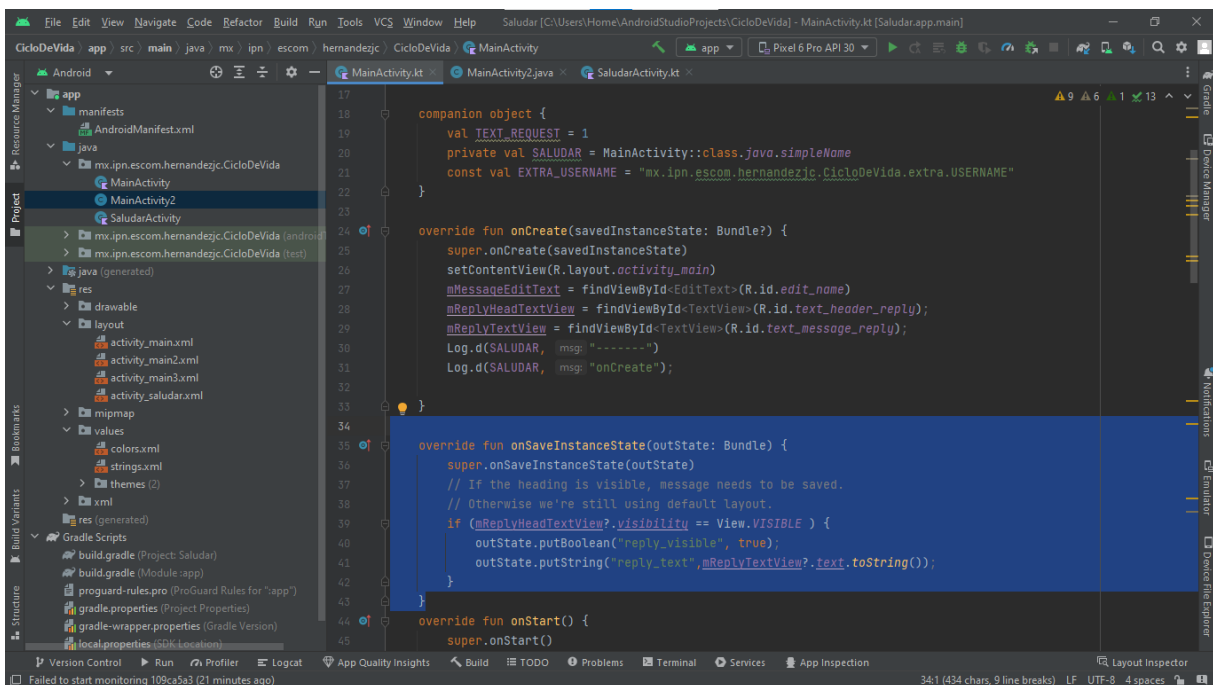
Android Studio interface showing the Logcat window. The top bar indicates the project is 'CicloDeVida' and the selected activity is 'MainActivity'. The Logcat window is filtered to show logs for 'package:mine.MainActivity' on a 'Xiaomi Redmi Note 7 (109ca5a3) Android 10, API 29'. The log entries show the lifecycle of the MainActivity, including onCreate, onStart, onResume, onPause, onStop, onDestroy, and onRestart. The log also shows the execution of the 'Slow Looper main' and the 'updateByCheckExcludeList' method. The log entries are as follows:

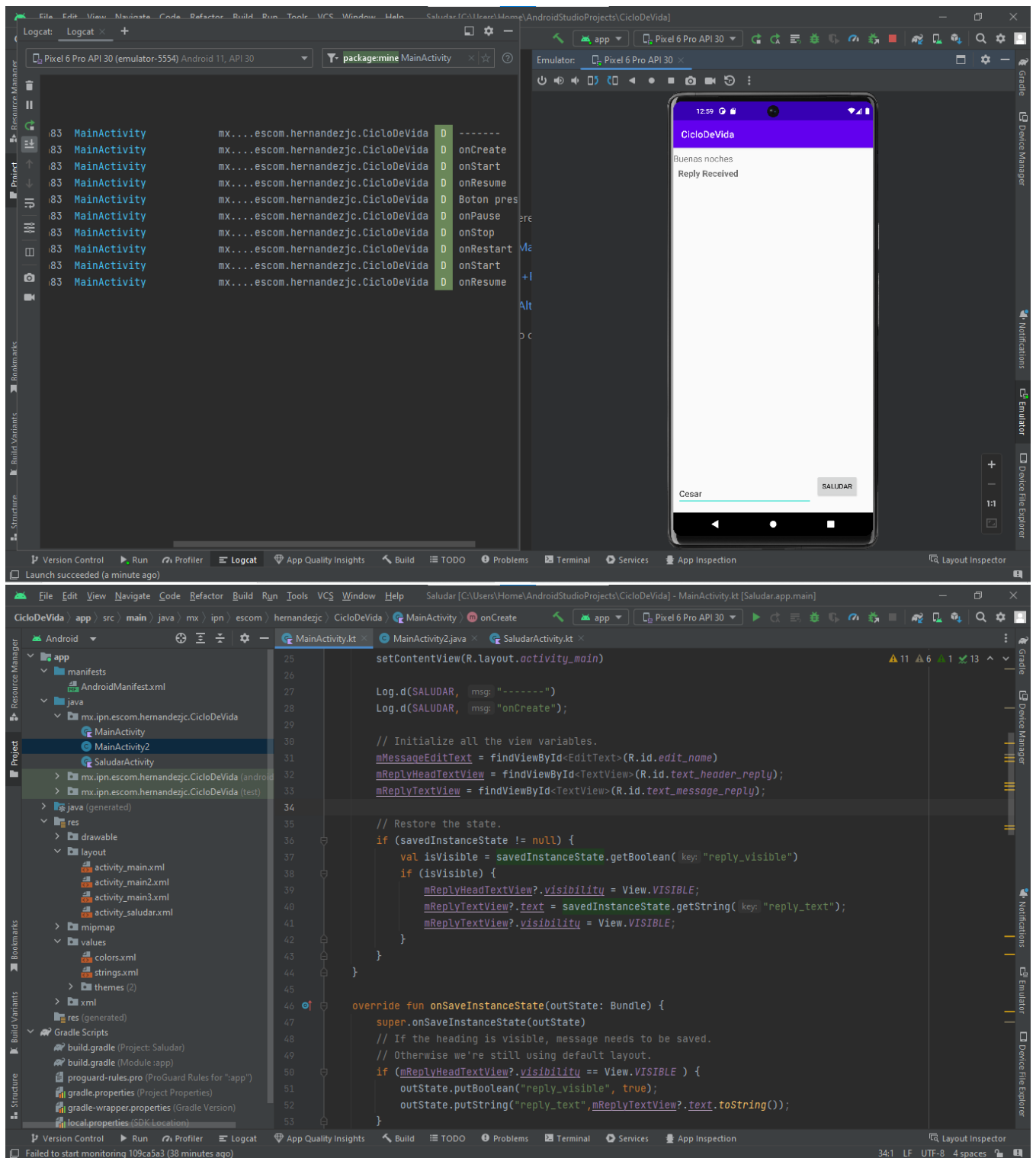
Time	Thread	Class	Message
2023-03-24 18:10:52.495	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:10:52.503	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D -----
2023-03-24 18:10:52.503	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onCreate
2023-03-24 18:10:52.507	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStart
2023-03-24 18:10:52.510	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onResume
2023-03-24 18:11:12.539	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onPause
2023-03-24 18:11:12.541	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStop
2023-03-24 18:11:12.544	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onDestroy
2023-03-24 18:11:12.614	1953-2969	InputDispatcher	system_server E Window handle Window{3152c0f u0 mx.ipn.escom.hernandezjc.CicloDeVida/.MainActivity} not found
2023-03-24 18:11:12.670	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:11:12.672	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:11:12.703	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:11:12.720	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:11:12.743	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D -----
2023-03-24 18:11:12.749	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onCreate
2023-03-24 18:11:12.753	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStart
2023-03-24 18:11:27.625	3229-3229	ForceDarkHelper	mx...escom.hernandezjc.CicloDeVida D onResume
2023-03-24 18:11:33.149	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D updateByCheckExcludeList: pkg: mx.ipn.escom.hernandezjc.CicloDeVida
2023-03-24 18:11:33.183	3229-3229	ActivityThread	mx...escom.hernandezjc.CicloDeVida W Boton presionado
2023-03-24 18:11:33.184	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida W SCHED: mx.ipn.escom.hernandezjc.CicloDeVida/.MainActivity [7]
2023-03-24 18:11:33.414	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onPause
2023-03-24 18:11:40.643	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStop
2023-03-24 18:11:40.644	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onRestart
2023-03-24 18:11:40.646	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onStart
2023-03-24 18:11:40.646	3229-3229	MainActivity	mx...escom.hernandezjc.CicloDeVida D onResume

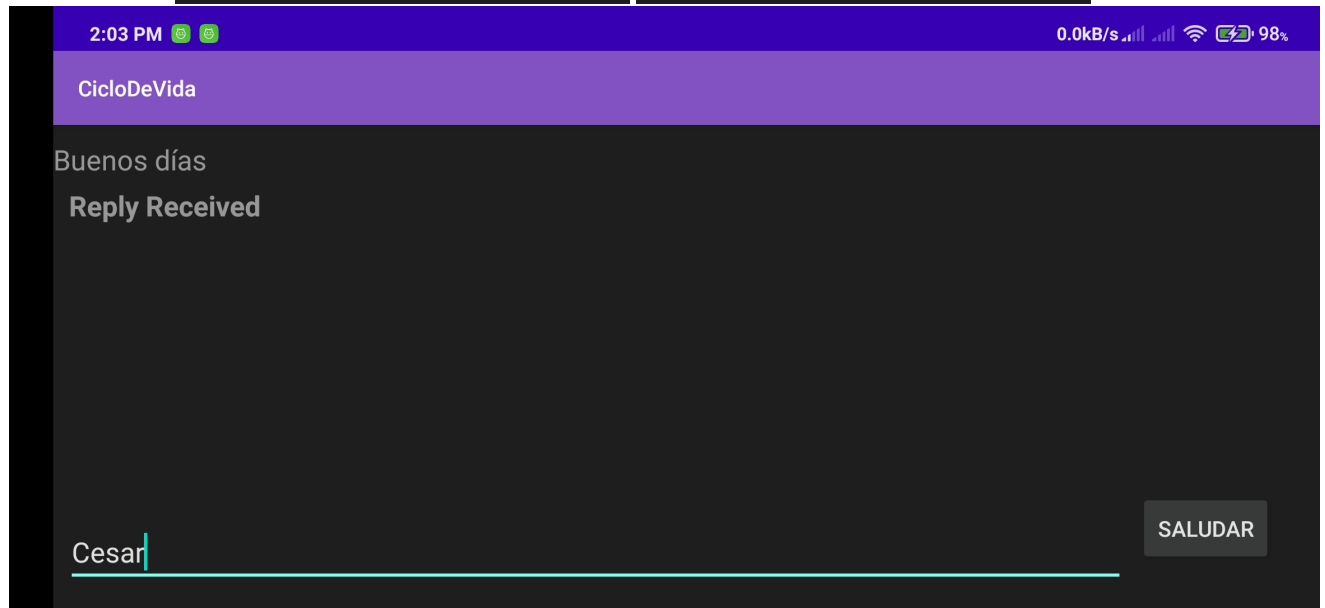
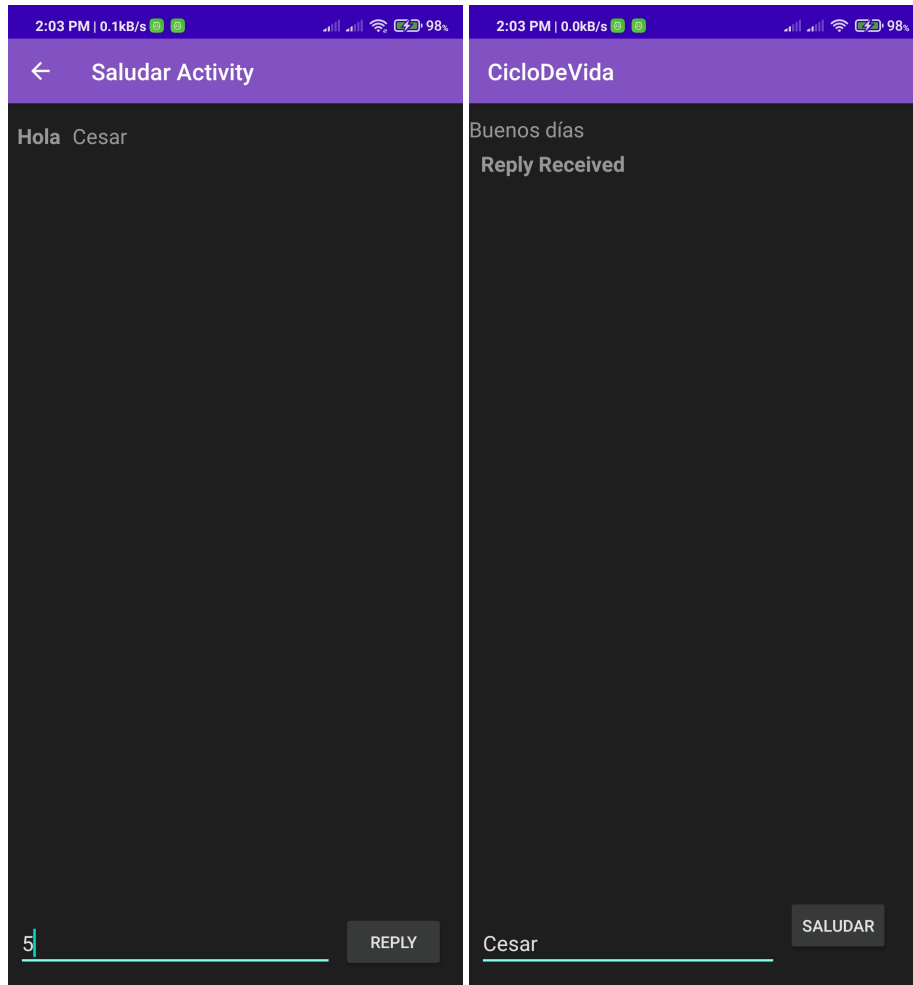


2.1.2. Task 2: Save and restore the Activity instance state

In this task i added code to preserve the instance state of the two TextView elements using onSaveInstanceState().





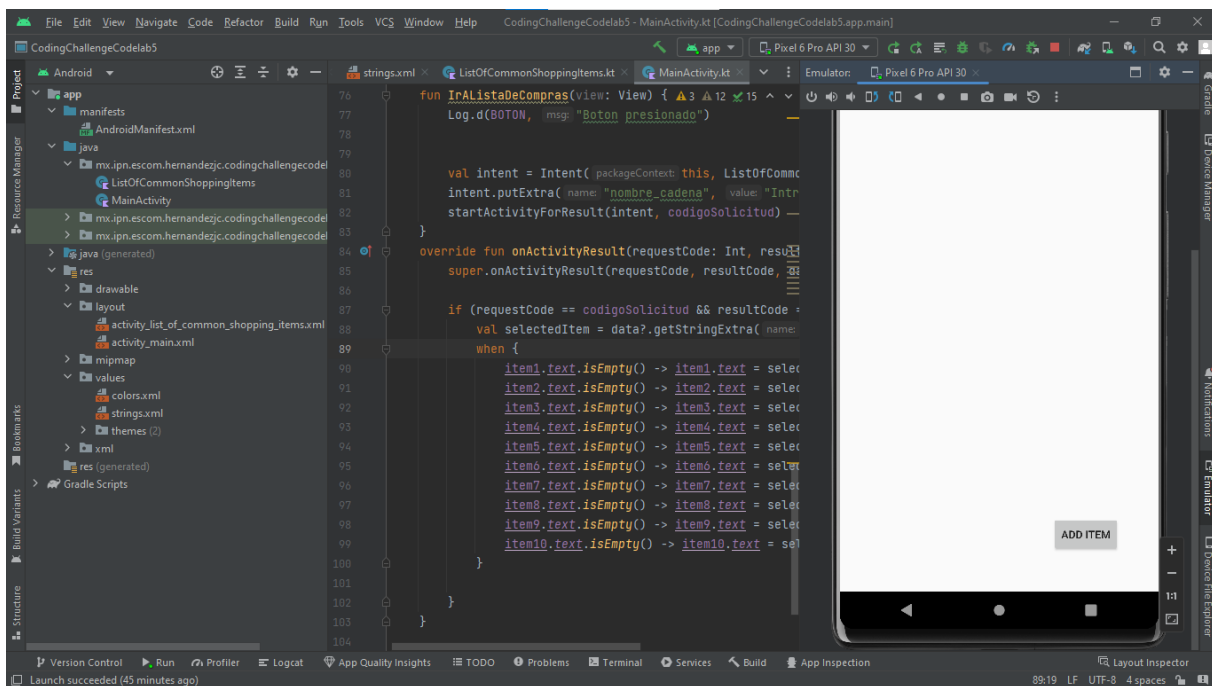


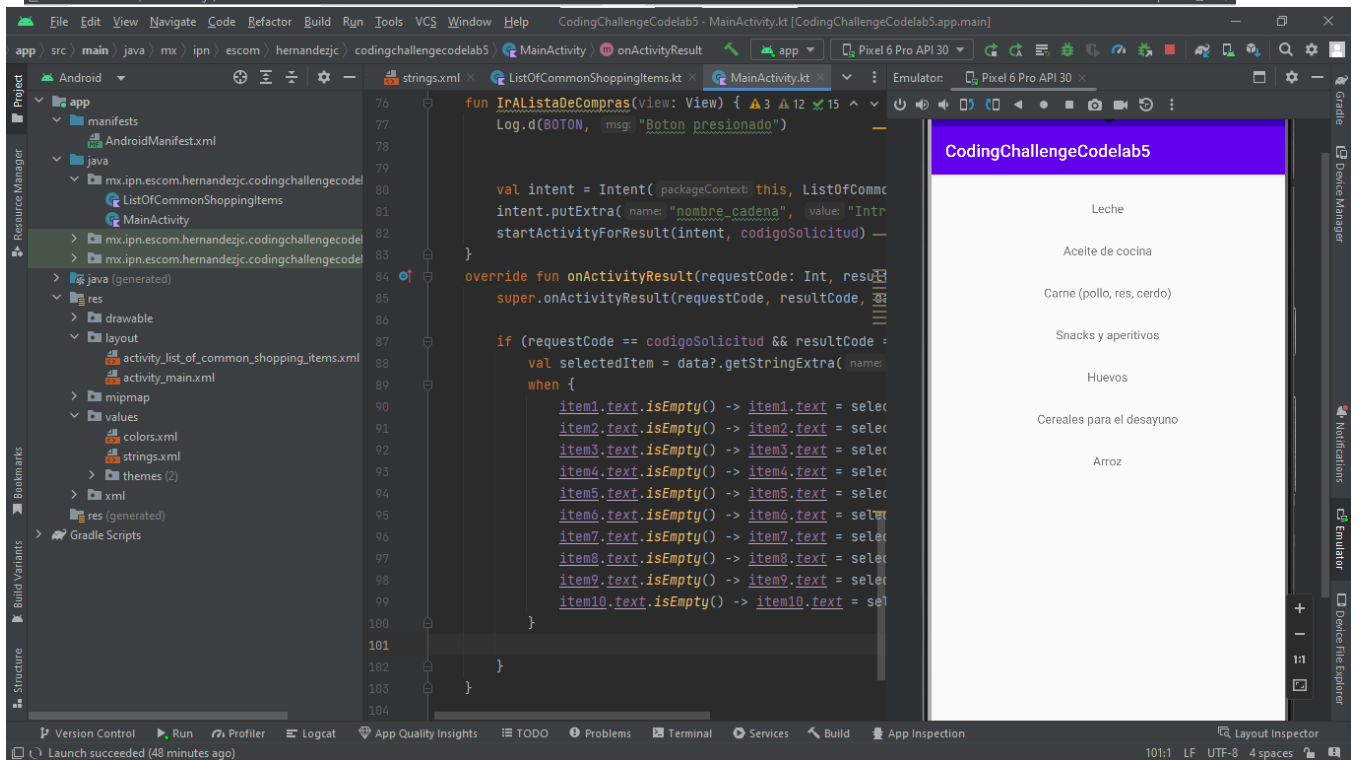
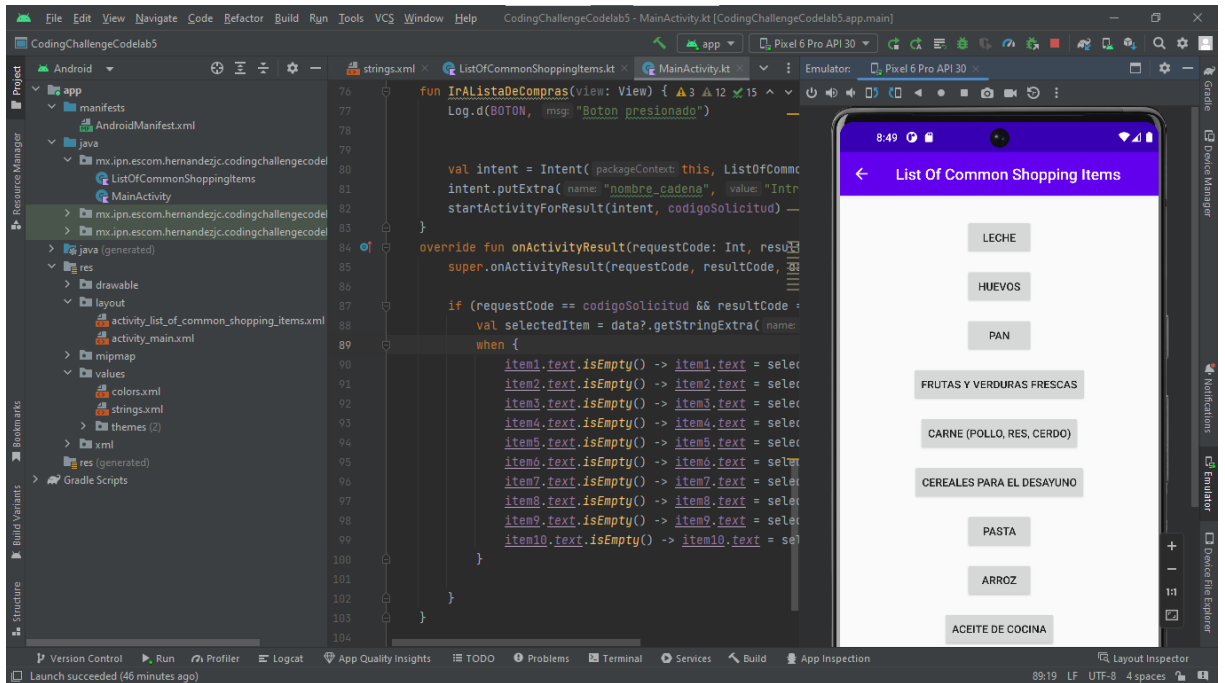
2.2. Coding Challenge

Challenge: Create a simple shopping-list app with a main activity for the list the user is building, and a second activity for a list of common shopping items.

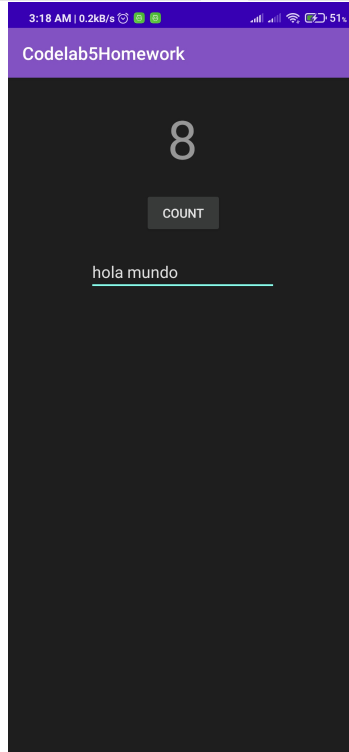
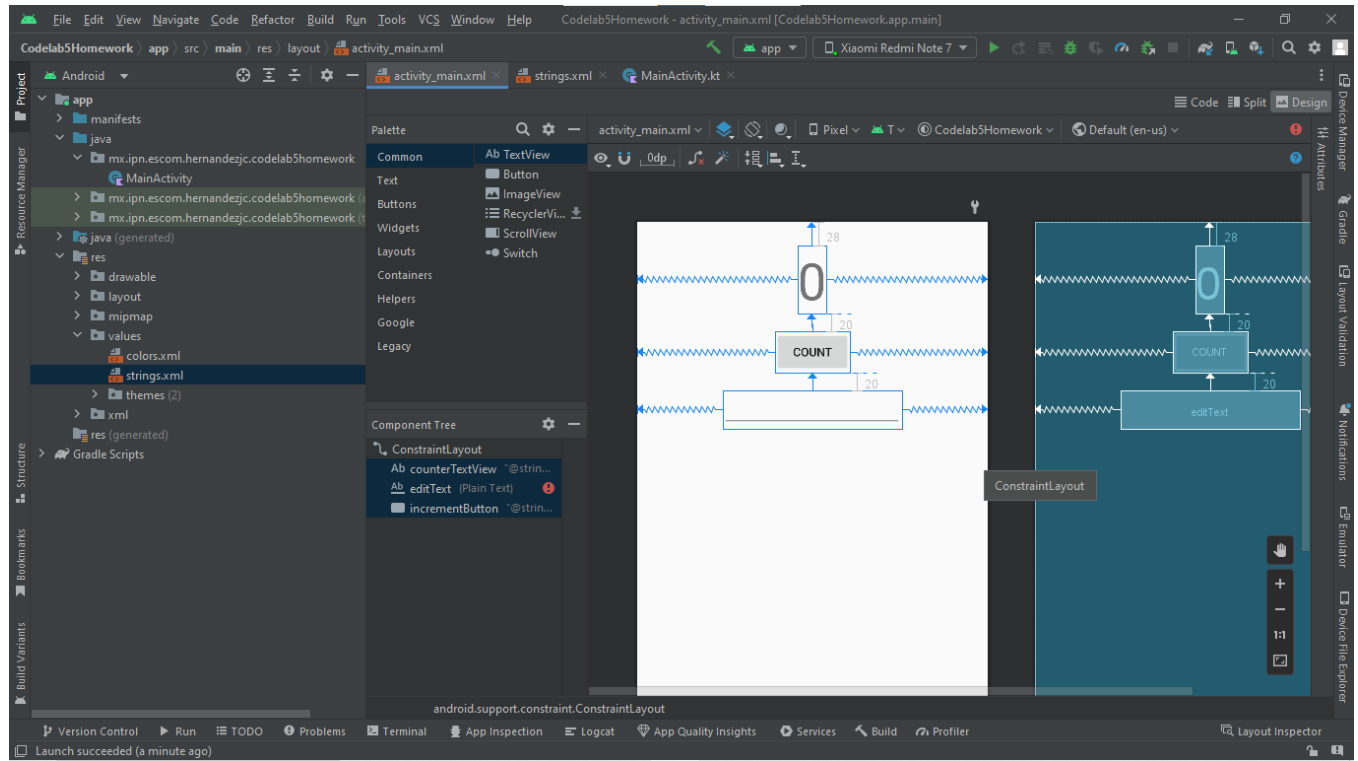
- The main activity should contain the list to build, which should be made up of ten empty TextView elements.
- An Add Item button on the main activity launches a second activity that contains a list of common shopping items (Cheese, Rice, Apples, and so on). Use Button elements to display the items.
- Choosing an item returns the user to the main activity, and updates an empty TextView to include the chosen item.

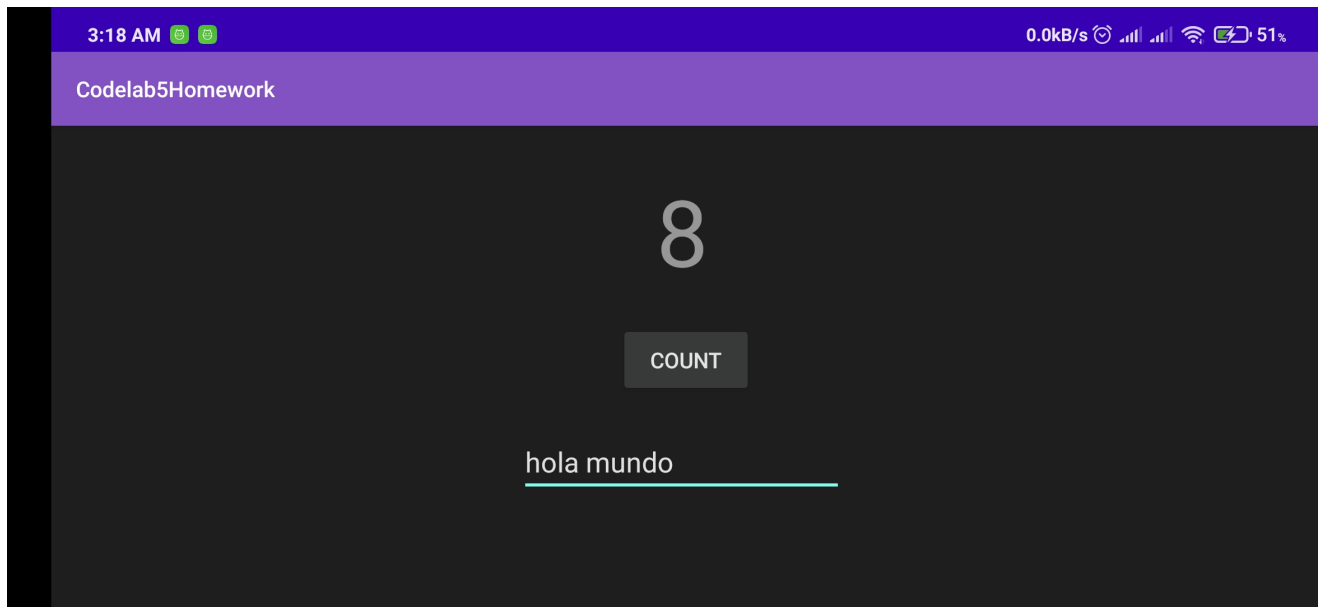
Use an Intent to pass information from one Activity to another. Make sure that the current state of the shopping list is saved when the user rotates the device.





2.3. Homework





Question 1

If you run the homework app before implementing `onSaveInstanceState()`, what happens if you rotate the device? Choose one:

R:

->The counter is reset to 0, but the contents of the EditText is preserved.

Question 2

What Activity lifecycle methods are called when a device-configuration change (such as rotation) occurs? Choose one:

R:

->Android shuts down your Activity by calling `onPause()`, `onStop()`, and `onDestroy()`, and then starts it over again, calling `onCreate()`, `onStart()`, and `onResume()`. Android immediately calls `onResume()`.

Question 3

When in the Activity lifecycle is `onSaveInstanceState()` called? Choose one:

R:

->`onSaveInstanceState()` is called before the `onCreate()` method.

Question 4

Which Activity lifecycle methods are best to use for saving data before the Activity is finished or destroyed? Choose one:

R:

->`onPause()` or `onStop()`