Mobile App Development

Continuous assessment Test 5 – CAT5

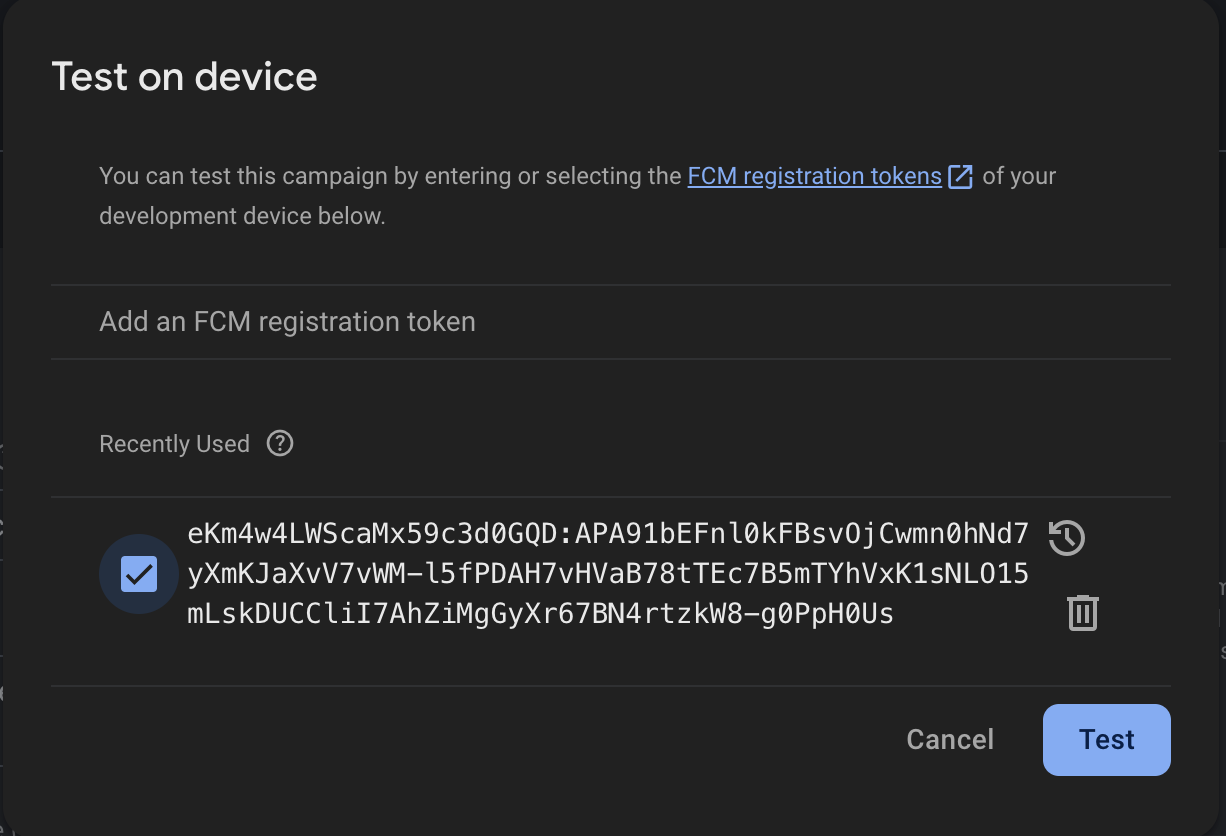
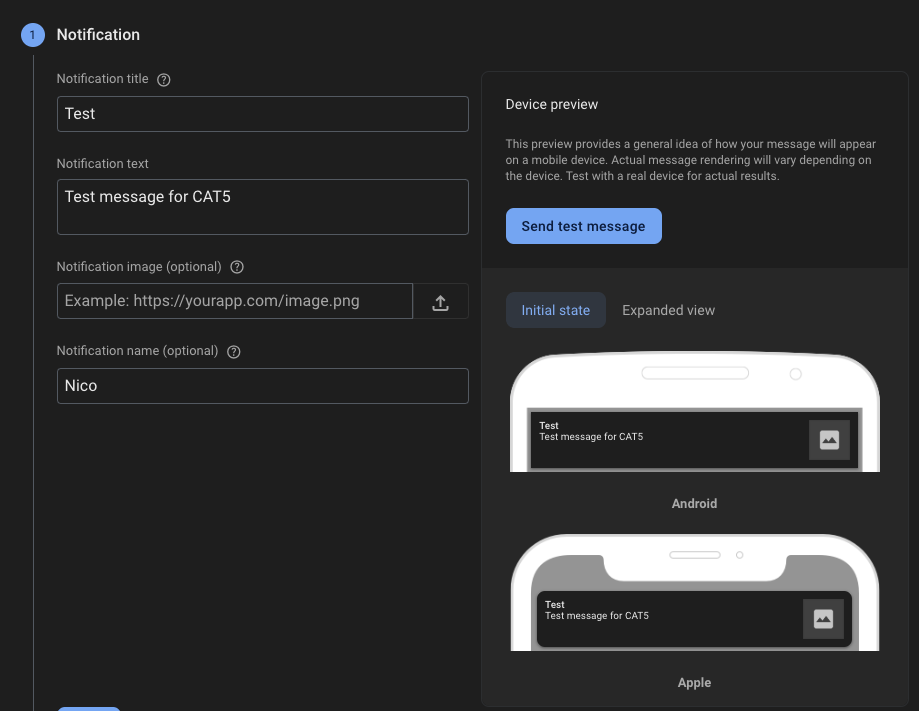
Nicolas D’Alessandro Calderon

1. Setting up remote notifications in Firebase
   1. Get the FMC token:





1.2 Build message in Firebase and send a test notification using the above obtained token:



1.3 Received notification:

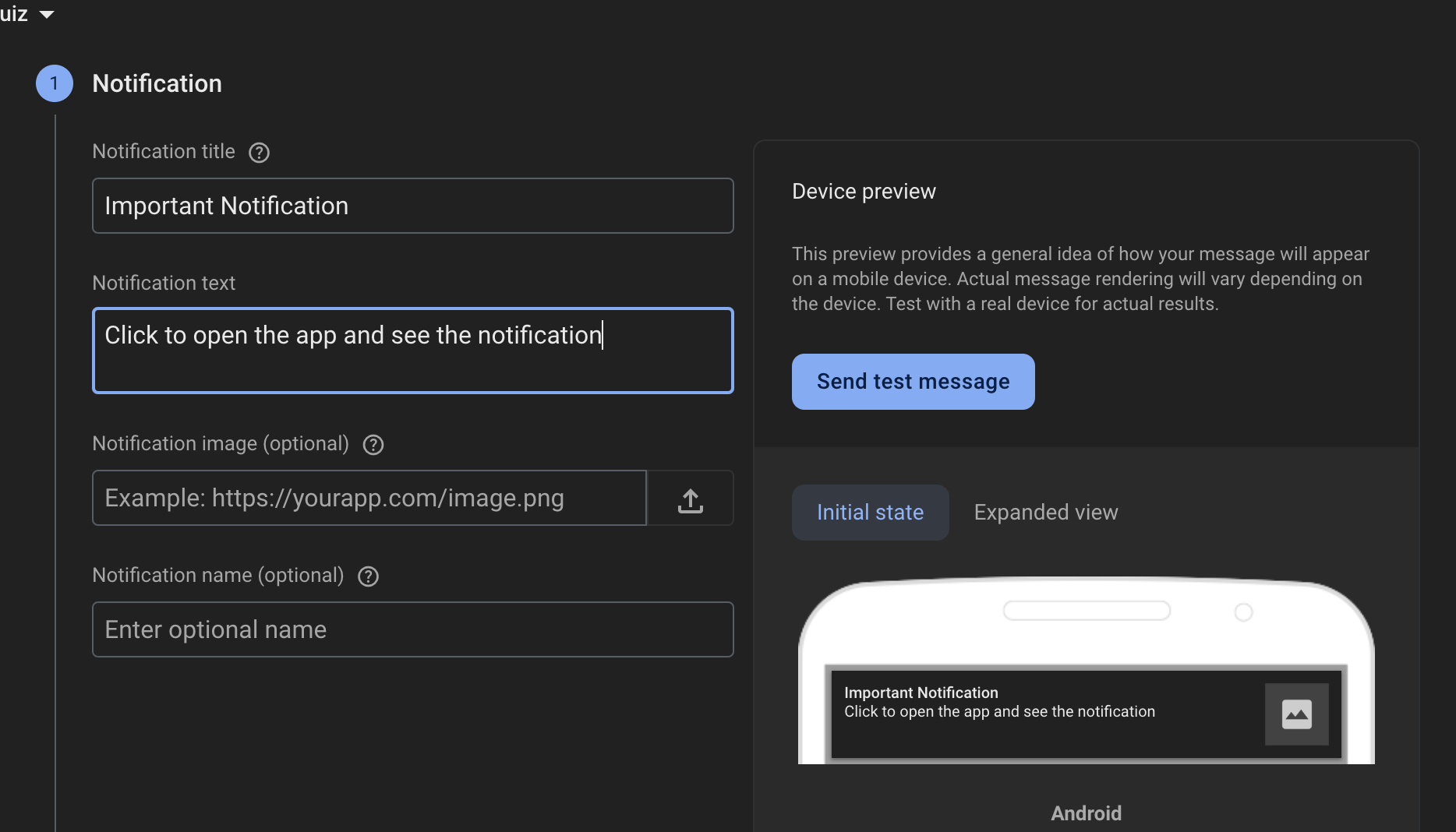
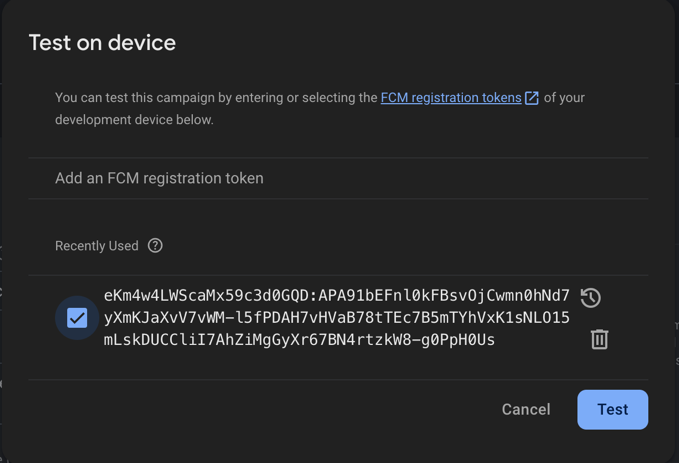
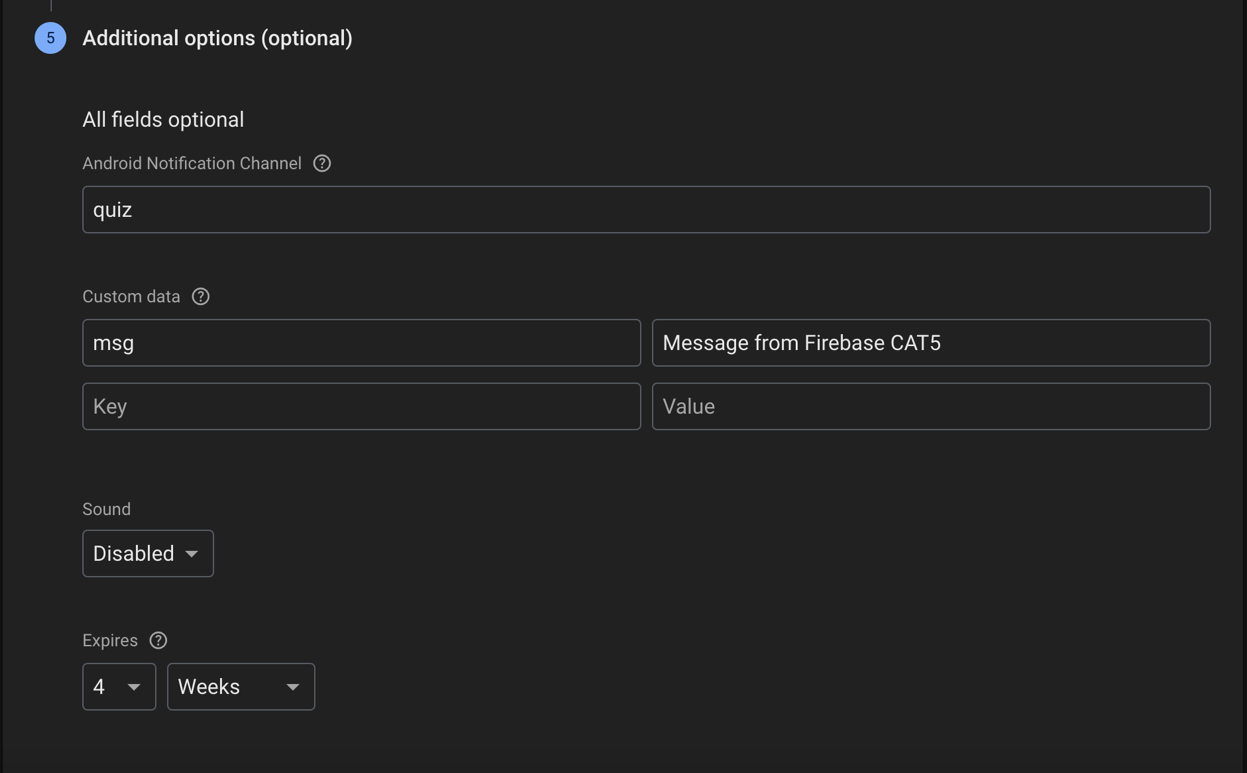
A screenshot of a phone

AI-generated content may be incorrect.



1. Opening an app when clicking on a notification

2.1 Build test notification with specific parameters and send new test notification:



2.2 Click on the received notification to see key value message created:

A screen shot of a phone

AI-generated content may be incorrect.A screen shot of a phone

AI-generated content may be incorrect.

1. Concurrent programming

3.1 If we run the first code we obtained the following error:

A screenshot of a computer

AI-generated content may be incorrect.

3.2 We can see in the code all the steps follow to reproduce the error and resolve it using concurrency:

override fun onMessageReceived(remoteMessage: RemoteMessage) {  
 val title = remoteMessage.*notification*?.*title* ?: "No title"  
 val body = remoteMessage.*notification*?.*body* ?: "No body"  
  
 // BEGIN-CODE-UOC-CAT5-3A  
 // Reproducing the error AlertDialog directly from background thread  
 /\*  
 val builder = AlertDialog.Builder(MainActivity.gmainActivity)  
 builder.setTitle(title)  
 builder.setMessage(body)  
 builder.setPositiveButton("OK") { dialog, \_ -> dialog.dismiss() }  
 builder.show()  
 \*/  
 // END-CODE-UOC-CAT5-3A  
  
 // BEGIN-CODE-UOC-CAT5-3B  
 // Why this fails:  
 // ERROR: CalledFromWrongThreadException.  
 // Android not allow updating the UI from background threads.  
 // The onMessageReceived is being executed in a background thread,  
 // and AlertDialogs must be created on the main thread.  
 // This is doing to crash the app.  
 // END-CODE-UOC-CAT5-3B  
  
 // BEGIN-CODE-UOC-CAT5-3C  
 // Correct using coroutine on main thread as explained in the wiki  
 GlobalScope.*launch*(Dispatchers.Main) **{** val builder = AlertDialog.Builder(MainActivity.gmainActivity)  
 builder.setTitle(title)  
 builder.setMessage(body)  
 builder.setPositiveButton("OK") **{** dialog, \_ **->** dialog.dismiss() **}**

builder.show()  
 **}**}  
 // END-CODE-UOC-CAT5-3C

3.3 Now, with the concurrency code applied, we build a new test message and ensure the notification is being extracting the title and the body and keeping the app in the main plane:

A screenshot of a device

AI-generated content may be incorrect.A screen shot of a phone

AI-generated content may be incorrect.A screenshot of a phone

AI-generated content may be incorrect.Screens screenshot of a black screen

AI-generated content may be incorrect.

1. Geolocation and maps

4.1 Permission requested to the user:

A screenshot of a phone

AI-generated content may be incorrect.

4.2 Coordinates obtained with FusedLocationProviderClient:



4.3 Theory answers:

// BEGIN-GEOLOCATION-3  
/\*  
 location.distanceTo will give the straight line distance in meters but  
 it not indicates the actual walking distance to the point.  
 It is useful for basic estimation in navigation, but if we want  
 the walking distance calculation, we will need the layout of the streets.  
\*/  
// END-GEOLOCATION-3  
  
// BEGIN-GEOLOCATION-4  
/\*  
 To get the accurate layout for calculating the walking distance, we need to  
 use a routing API like Google Maps Directions API that supports the walking mode.  
 This API considers streets and constrains needed for walking distance calculation.  
 As the statement mentions, it is a paid service and requires billing enabled  
 in Google Cloud.  
\*/  
// END-GEOLOCATION-4