CAT5

Presentation

In this Continuous Assessment Test (CAT), you will learn how to include extra features in an app such as notifications, positioning and maps, using external libraries and APIs.

Competencies

This CAT will develop the following competencies of the Bachelor's degree in Techniques for Software Development:

- Adapt to new software development technologies and to future environments, updating professional skills.
- Design and build computer applications using development, integration and reuse techniques.

Objectives

The learning outcomes of this CAT are the following:

- Code using a programming language for developing native mobile apps (Kotlin).
- Add notifications to a mobile app.
- Use external libraries and APIs for adding features to a mobile app.
- Learn external libraries and APIs dealing with maps and positioning information.

Problem statement

In this continuous assessment test, we will practice using several features in our app: notifications, concurrent programming and back-end monitoring systems.

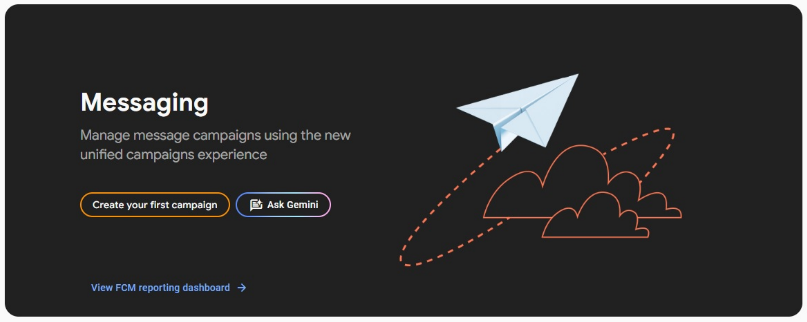
To implement notifications and monitoring, we will continue to use the Firebase back-end. We will use the same Firebase project that we already created in the previous CAT. However, it will be necessary to download the `google-services.json` file again (we will provide more instructions about this later).

Remember that, in our app, the username and password needed to log in are the following:

```
user:   user1@uoc.com
pwd:    123456
```

1. **Setting up remote notifications in Firebase** (Weight: 25%)

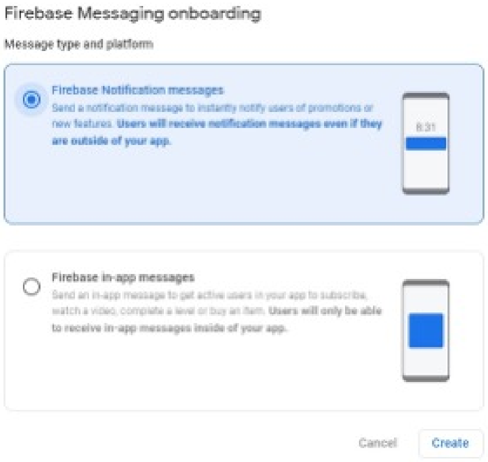
   (a) Access the Firebase project "`SeminarData`" that you created in the previous CAT. There, click on option "`Run / Messaging`" in the left menu to setup the notification for this project.



Messaging

Manage message campaigns using the new unified campaigns experience

   Then, click on the "`Create your first campaign`" button.



Firebase Messaging onboarding

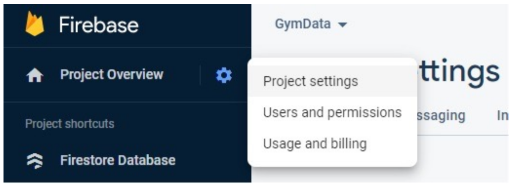
Message type and platform

● Firebase Notification messages
Send a notification message to instantly notify users of promotions or new features. **Users will receive notification messages even if they are outside of your app.**

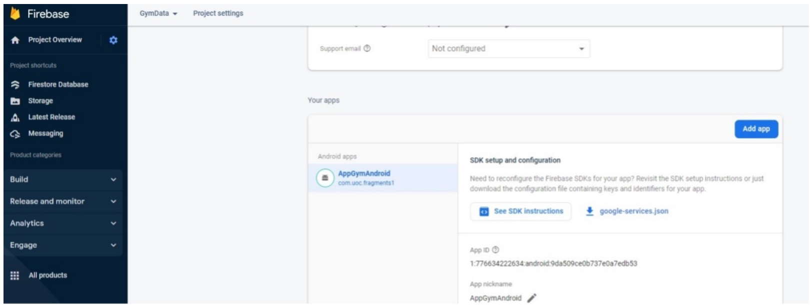
○ Firebase in-app messages
Send an in-app message to get active users in your app to subscribe, watch a video, complete a level or buy an item. **Users will only be able to receive in-app messages inside of your app.**

Select the first option ("Firebase Notifications messages") to also receive notifications when the app is not running.

(b) Download the google-services.json file again from the top of the left menu. To do that, click the nut icon and click "Project Overview / Project settings".



Scrolling down a bit on the Project settings screen and you will see the option to download the file google-services.json for this project.



Support email: Not configured

Your apps

Android apps: AppGymAndroid, com.uoc.fragments1

SDK setup and configuration

Need to reconfigure the Firebase SDKs for your app? Revisit the SDK setup instructions or just download the configuration file containing keys and identifiers for your app.

App ID: 1:776634222634:android:9da509ce0b737e0a7edb53

App nickname: AppGymAndroid

Then copy this file into the app folder.

Important: Edit the `build.gradle` (Module: fragments1.app) and remove the comment from the plugin with id the following id: '`com.google.gms.google-services`'.

Follow the instruction from chapter *Remote notifications* (Section 13.1) of the course wiki to add the code necessary to get the notification token.

Some considerations about the Theoretical material:

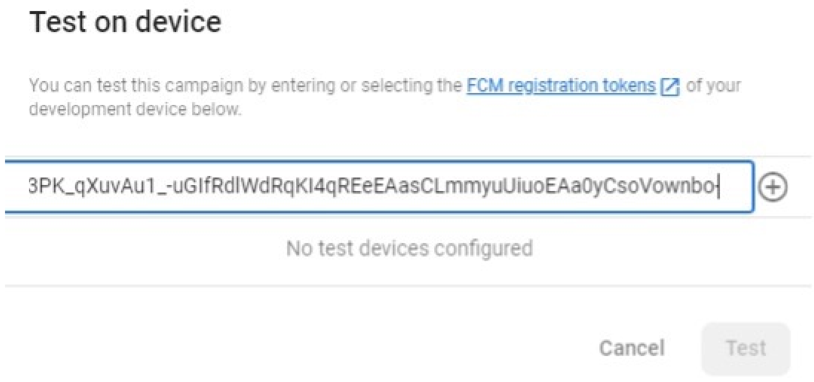
i. The class `MyFirebaseMessagingService` should be created directly in `com.uoc.pr1`

ii. Do not write this code: `sendRegistrationToServer(token)`.

iii. You can get the token in `addOnCompleteListener` of `FirebaseMessaging.getInstance()` in the `onCreate` callback of our main Activity

iv. You must move the App to the background before sending a message from Firebase. We have not yet implemented the reception of a message when the App is in the foreground.

(c) Now with this token you can send messages that will become notifications from Firebase. You need to go back to the point where you left the Firebase console at the end of step 1.a) If you don't have that window active in the browser, simply repeat step 1.a). Then, type a title and text and click on the "`Send test message`" button. Finally, add the token of our device (the one we located in the previous step).



Test on device

You can test this campaign by entering or selecting the FCM registration tokens of your development device below.

3PK_qXuvAu1_-uGIfRdlWdRqKI4qREeEAasCLmmyuUiuoEAa0yCsoVownbo

No test devices configured

Now you can try to send this notification as a test.

Important: Each time we install/uninstall an app we will receive a different token.

(d) Include a screenshot of the notification in your submission.

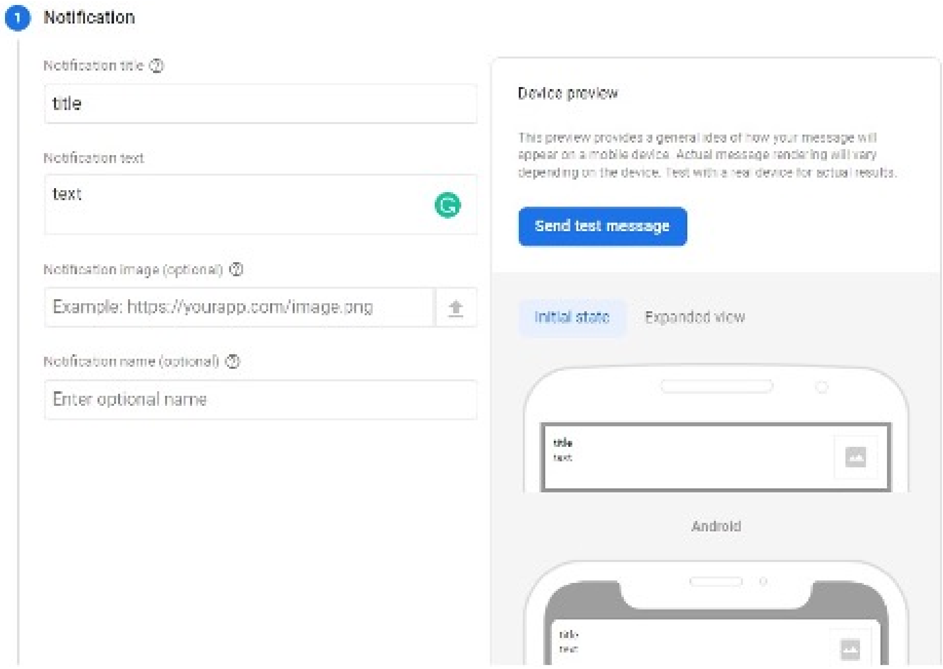
2. **Opening an app when clicking on a notification** (Weight: 20%)

   In the method `onCreate` of the class `MainActivity`, add the code to detect if the application has been opened by clicking a notification. If this is true, then show an alert with the value of the `msg` key.

   Remember that you will need to include two snippets of code that are presented in the course wiki: specifying the notification channel (quiz) and checking the `intent.extras`.

   This time you will need to set up a campaign in Firebase in its entirety in order to assign the channel.

   Please follow this list of steps:



1 Notification

Notification title: title

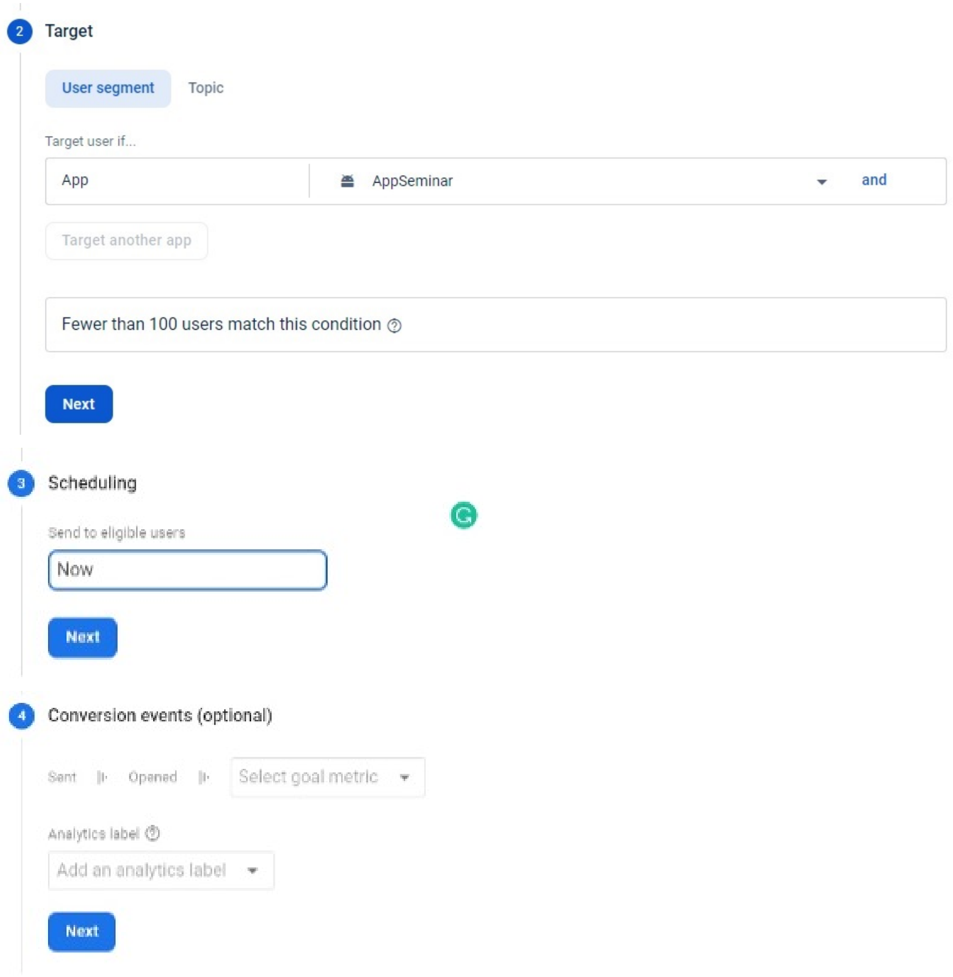
Notification text: text

Notification image (optional): Example: https://yourapp.com/image.png

Notification name (optional): Enter optional name

Device preview

This preview provides a general idea of how your message will appear on a mobile device. Actual message rendering will vary depending on the device. Test with a real device for actual results.



2 Target

User segment | Topic

Target user if...

App: AppSeminar, and

Fewer than 100 users match this condition

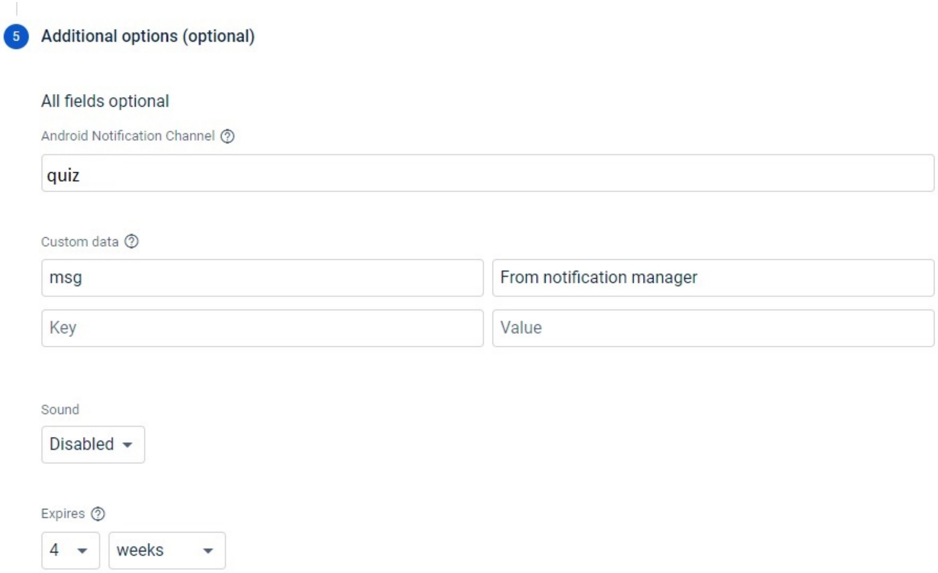
3 Scheduling

Send to eligible users: Now

4 Conversion events (optional)

Sent · Opened · Select goal metric

Analytics label: Add an analytics label



5 Additional options (optional)

All fields optional

Android Notification Channel: quiz

Custom data: msg — From notification manager

Key — Value

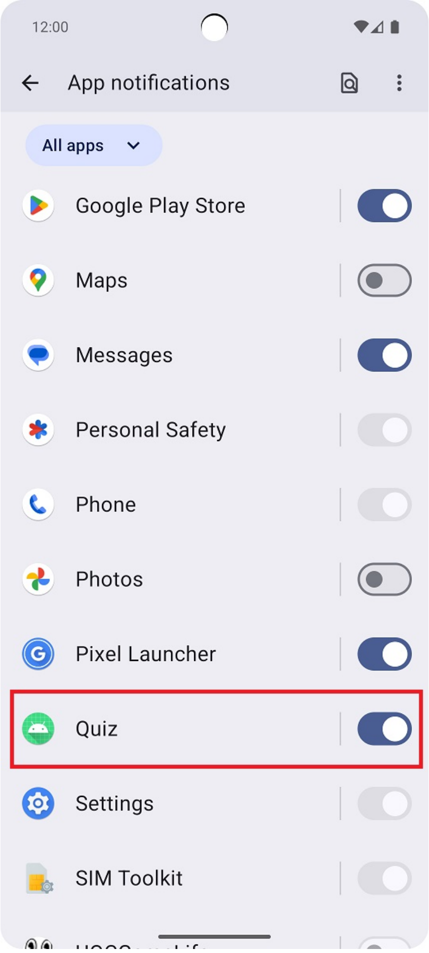
Sound: Disabled

Expires: 4 weeks

Once step 5 has been configured with the key "`msg`" and the value "`message from Firebase`" we go back to step 1 and open it and use "`Send test message`" to do the tests.

To test the notification we need to move the app to the background on our emulator or on the physical device where we are performing the test. We will see the notification appear at the top of our device in the notifications center.

Remember to go to the Settings / Notifications section of the emulated system to enable notifications, as shown in the following image.



12:00

App notifications

All apps

- Google Play Store
- Maps
- Messages
- Personal Safety
- Phone
- Photos
- Pixel Launcher
- Quiz
- Settings
- SIM Toolkit

3. **Concurrent programming** (Weight: 25%)

   Modify the code from onMessageReceived funtion to display an alert with the notification title and text.

   (a) Modify the code from onMessageReceived method to display an alert with the notification title and text. When using AlertDialog.Builder you can use as a parameter the static property called MainActivity.gmainActivity that gives you access to the main activity. You will need to study the User interface section (Section 5) of the course wiki. **This code will cause an error**. Use breakpoints and the Logcat console to indicate the error in a comment at that code point.

   (b) Answer the following theoretical question in a comment (5 lines): Why did that error occur?

   (c) To fix this error, you should use the information provided about concurrent programming in the Concurrency section (Section 10, GlobalScope.launch(Dispatchers.Main)) of the course wiki.

4. **Geolocation and maps** (Weight: 30%)

   Let's add the code to ask for the device geolocation at the end of MainActivity:OnCreate:

   - Add the necessary code to ask the ACCESS_FINE_LOCATION permission to the user. Uncomment the content between //BEGIN-GEOLOCATION-1 and //END-GEOLOCATION-1. Copy the methods isPermissionsGranted, requestLocationPermission, and onRequestPermissionsResult (Section 14.1 from the course wiki). Remember to call the method GetLocation() in the branch that accepts permissions in onRequestPermissionsResult.
   - Now, you should fill a request for the current geolocation using getFusedLocationProviderClient within the BEGIN-GEOLOCATION-2 comments found inside the GetLocation method. You should use the following code:

```
fusedLocationClient.getCurrentLocation(
    Priority.PRIORITY_HIGH_ACCURACY,  // High accuracy mode
    null // No CancellationToken needed
).addOnSuccessListener { location ->
    if (location != null) {
        // Use location
        val latitude = location.latitude
        val longitude = location.longitude
        Log.d("Location", "Latitude: $latitude, Longitude: $longitude")
    } else {
        // Handle null location
        Log.d("Location", "Location is null. Try again.")
    }
}.addOnFailureListener { e ->
    // Handle failure
    Log.e("LocationError", "Failed to get location", e)
}
```

- Within BEGIN-LOCATION-3, answer the following theoretical question in 5 lines: Suppose we use location.distanceTo to calculate the distance between your current position, obtained in the previous section, and a nearby point in our city. Does this distance indicate the walking distance to that place?
- Within BEGIN-LOCATION-4, answer the following theoretical question in 5 lines:
  Now, you must Investigate. What would we need to use to accurately calculate the walking distance between two points within a city?
  Hint:It is a paid service.

Resources

Basic Resources

- Course Wiki: Section 13 - Notifications
- Course Wiki: Section 14 - Geolocation and maps

Additional Resources

- Official Android developer documentation
- Official Android Studio documentation
- Maps SDK for Android
- Notifications in Android

Assessment criteria

- All exercises in this CAT must be solved **individually**.
- Each exercise has the following weight:
  1. 25%
  2. 20%
  3. 25%
  4. 30%
- The use of artificial intelligence tools is not allowed in this activity.

  The course plan and the UOC's site on academic integrity and plagiarism have information on what is considered misconduct in assessment activities and the consequences this may have.

Submission format and deadline

You must submit a ZIP file including the project with all changes and a PDF file with screenshots of the steps you followed to configure Firebase. The name of the file you submit must be **CAT5_SurnameName.zip**. This document must be submitted in the **Continuous Assessment Record** of your classroom **before 23:59 of 2025/06/01**. **Late submissions will not be accepted.**