

Guía de ejercicios - Distribución del aplicativo Android (II)



¡Hola! Te damos la bienvenida a esta nueva guía de estudio.

¿En qué consiste esta guía?

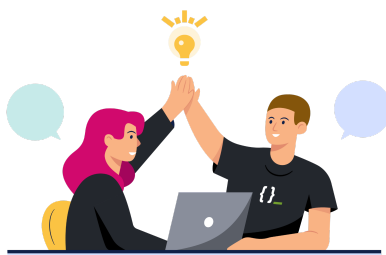
La siguiente guía de estudio tiene como objetivo practicar y ejercitar los contenidos que hemos visto en clase.

¡Vamos con todo!



Tabla de contenidos

Actividad guiada: Capturar crash usando Crashlytics	2
¡Manos a la obra! - Nombre de la actividad	15
Respuesta:	16
Preguntas de proceso	17
Preguntas de cierre	17
Referencias bibliográficas	17



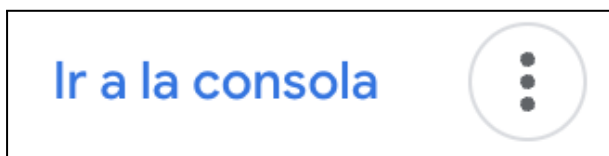
¡Comencemos!



Actividad guiada: Capturar crash usando Crashlytics

En la siguiente actividad crearemos un proyecto, generaremos un crash y lo capturaremos con Crashlytics. Esta actividad requiere que el estudiante tenga una cuenta en Firebase, en caso que no la tengas, puedes ir a <https://firebase.google.com/> e iniciar sesión con tu cuenta de Google.

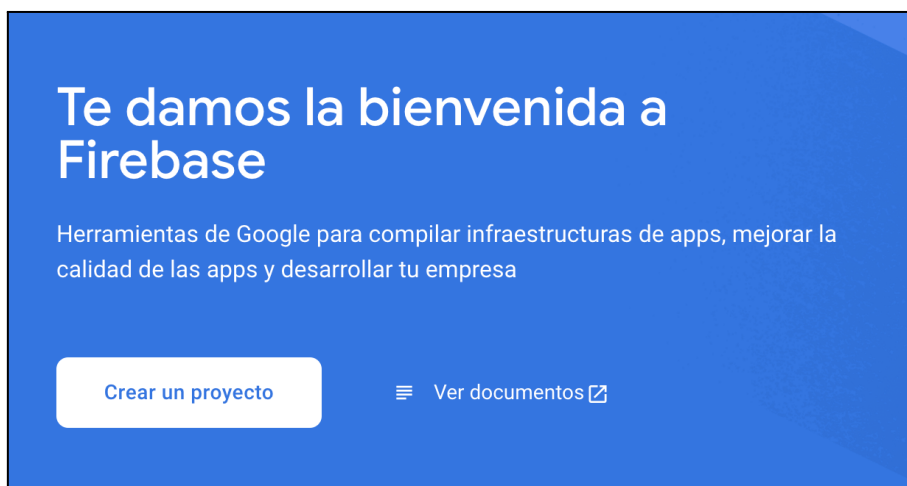
Una vez que hayas iniciado sesión, ve a **“ir a la consola”**, botón que se encuentra en la parte superior derecha de la página. Ahí podrás crear tu proyecto.



A continuación crearemos un proyecto en Android Studio en el que crearemos una app que tiene como finalidad generar un crash el que capturaremos con Firebase Crashlytics.

Sigue estos pasos:

- Crear un proyecto en Android Studio, selecciona una app con solo un Activity.
- Ahora en el lado de Firebase, crea un proyecto.



- Dale un nombre a tu proyecto:

Let's start with a name for your project [?]

Project name

Crash Latam

 crash-latam


Continue


- Puedes seleccionar habilitar Google Analytics para tu proyecto (es opcional, pero se recomienda hacerlo).


Google Analytics for your Firebase project


Google Analytics is a free and unlimited analytics solution that enables targeting, reporting, and more in Firebase Crashlytics, Cloud Messaging, In-App Messaging, Remote Config, A/B Testing, and Cloud Functions.


Google Analytics enables:


 A/B testing [?]

 Crash-free users [?]

 User segmentation & targeting across
Firebase products [?]

 Event-based Cloud Functions triggers [?]

 Free unlimited reporting [?]

 Enable Google Analytics for this project
Recommended

[Previous](#)

Continue

- Selecciona el nombre de la cuenta de Google Analytics. Si no existe una que haya sido creada previamente para este proyecto, la puedes crear ahora seleccionando **"Crear nueva"**.

Configure Google Analytics

Choose or create a Google Analytics account ⓘ

Select an account ▼

Upon project creation, a new Google Analytics property will be created in your chosen Google Analytics account and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. [Learn more](#) ⓘ.

[Previous](#) [Create project](#)

- Dale un nombre a la cuenta de Google Analytics y presiona **"Guardar"**.

Configure Google Analytics

New Google Analytics account name

crash-latam

Cancel [Save](#)

Analytics terms of service. [Learn more](#) ⓘ.

[Previous](#) [Create project](#)

- Luego revisa que todos los datos ingresados estén correctos, si es así, entonces presiona **"Crear proyecto"**.

Choose or create a Google Analytics account ?

crash-latam ▼

Analytics location ?

Chile ▼

Google Analytics is a business tool. Use it exclusively for purposes related to your trade, business, craft, or profession.

Data sharing settings and Google Analytics terms

☒ Use the default settings for sharing Google Analytics data. [Learn more](#)

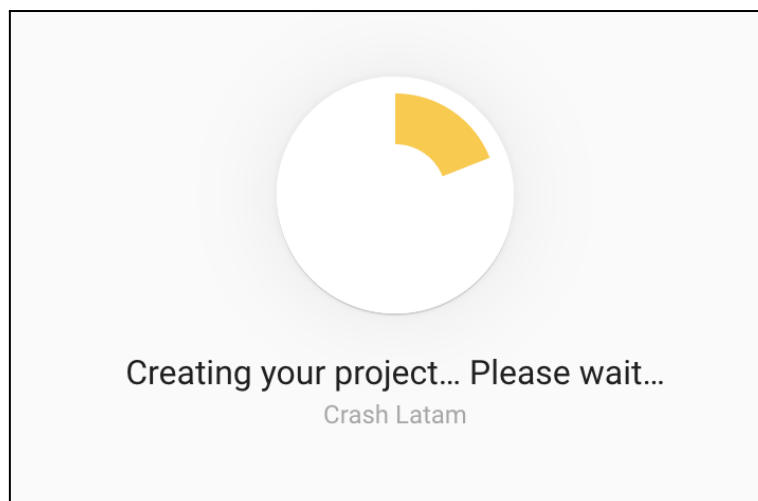
- ☐ Share your Analytics data with Google to improve Google Products and Services
- ☒ Share your Analytics data with Google to enable Benchmarking
- ☒ Share your Analytics data with Google to enable Technical Support
- ☒ Share your Analytics data with Google Account Specialists

☒ I accept the [Google Analytics terms](#)

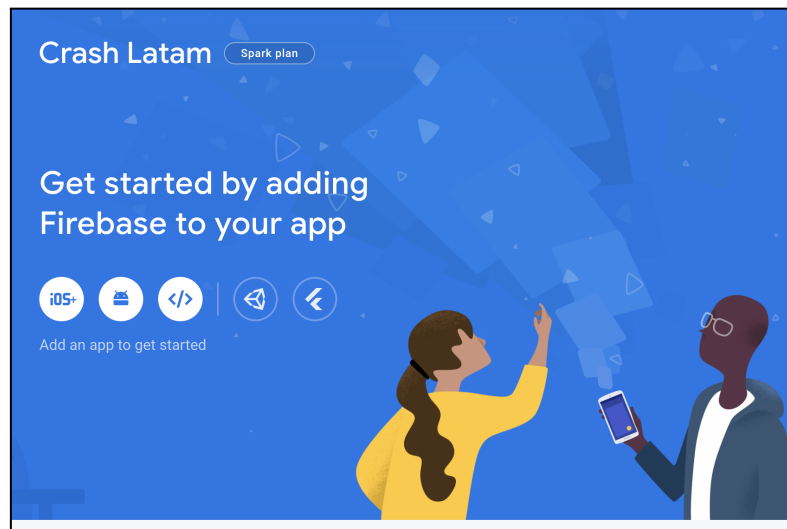
Upon project creation, a new Google Analytics property will be created and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. [Learn more](#)

[Previous](#) [Create project](#)

- La creación de tu proyecto en Firebase iniciará, esto puede tomar unos segundos:

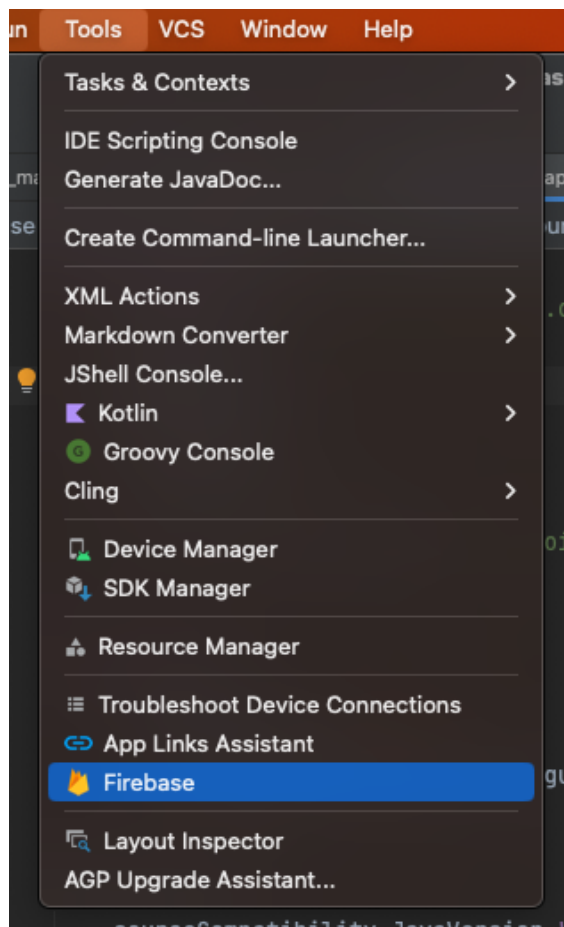


Una vez que el proceso haya terminado verás algo así:



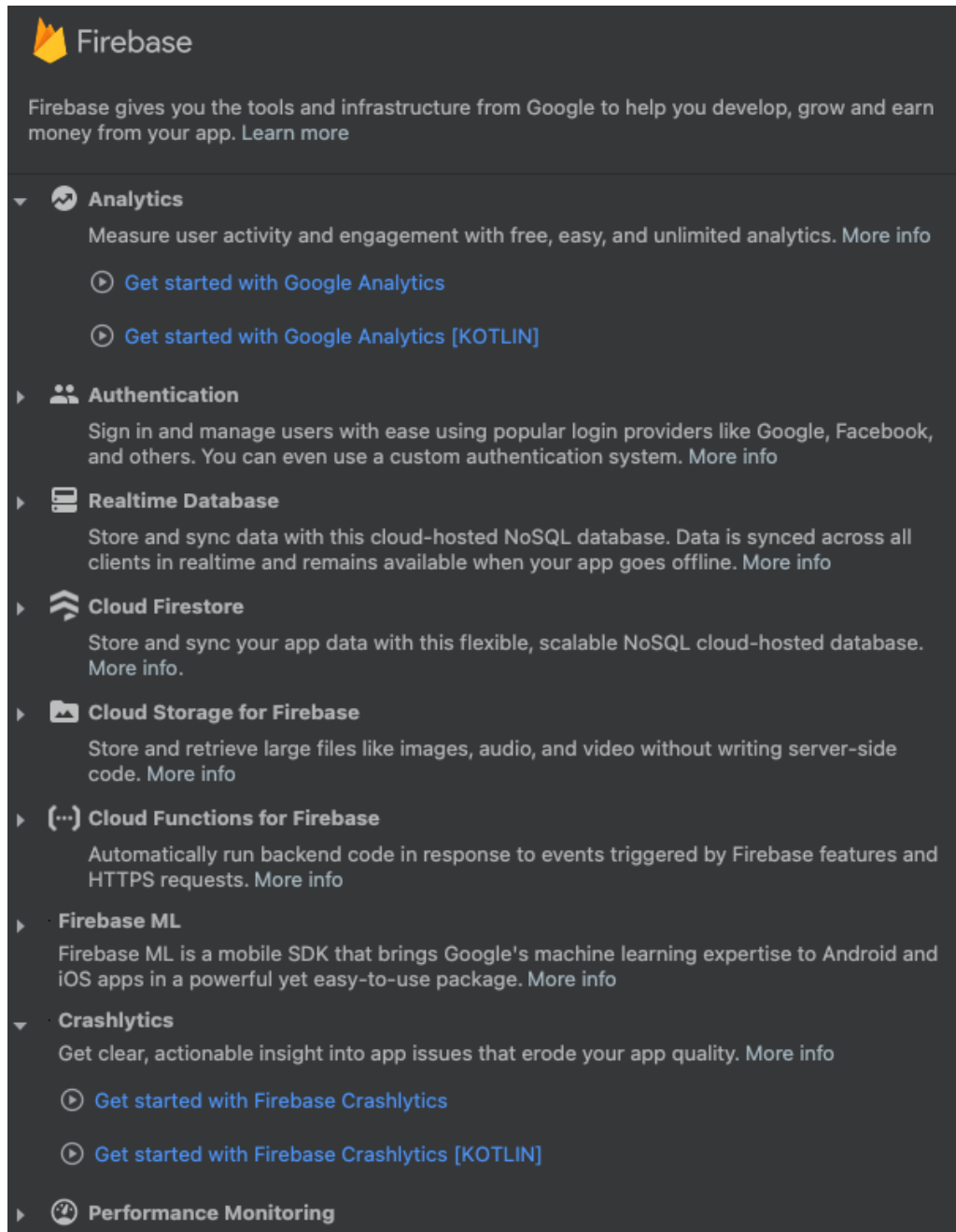
Ahora continuamos con Android Studio, necesitamos conectar el proyecto recién creado en Firebase con tu app en Android Studio, la forma más fácil de hacer esto es con el asistente de Android Studio.

Ve a Herramientas > Firebase



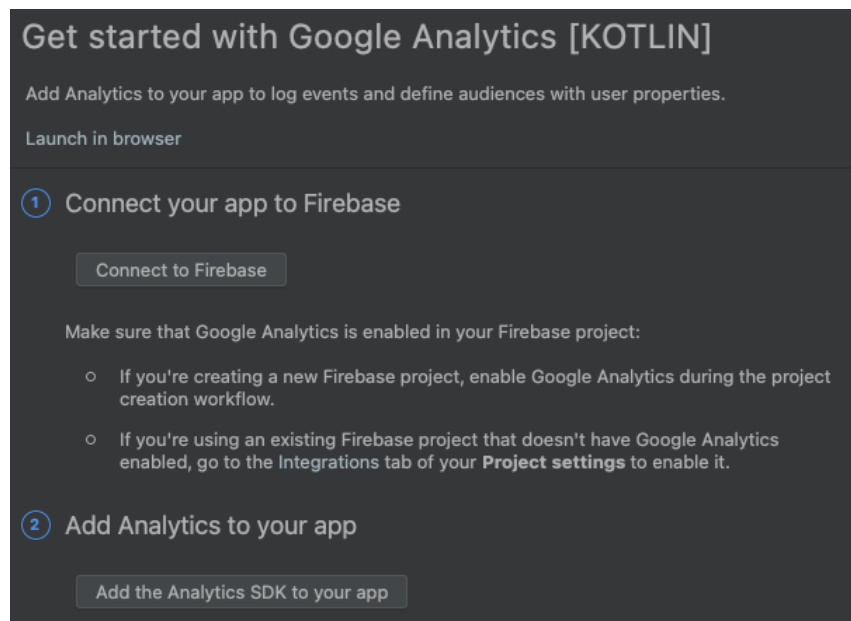
El asistente nos mostrará un menú en el cual debemos seleccionar:

- Analytics
- Crashlytics

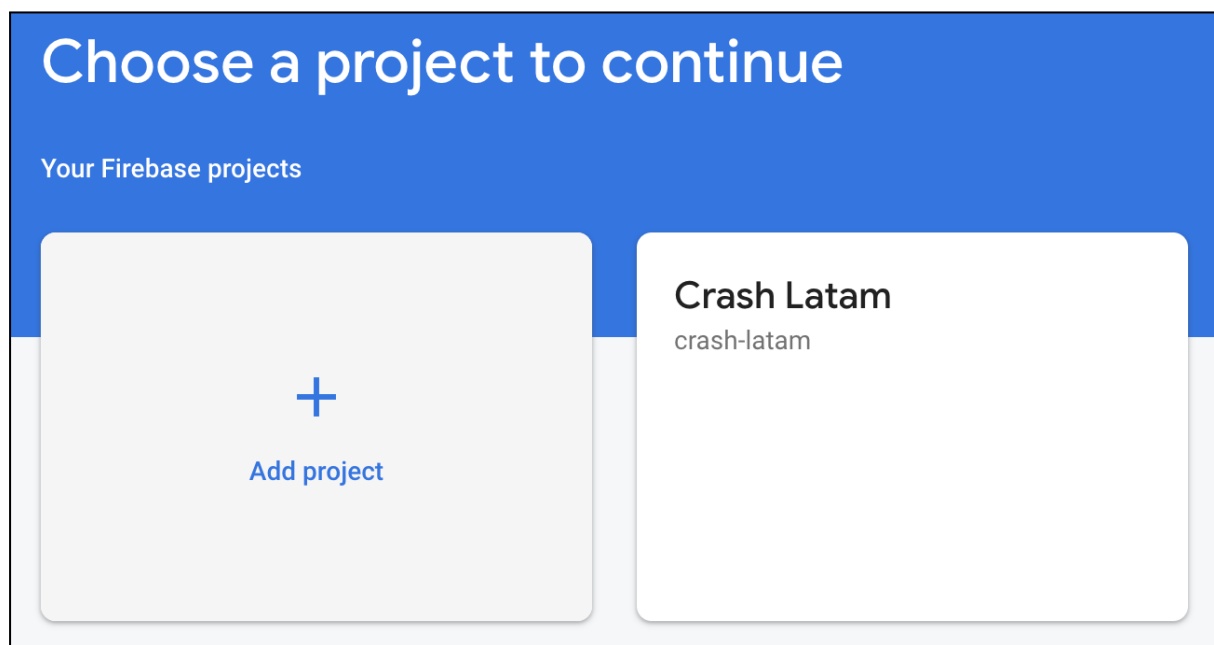


Asegúrate de seleccionar las opciones que dice "Kotlin", al presionar "Get started with Google Analytics" veremos las siguientes opciones:

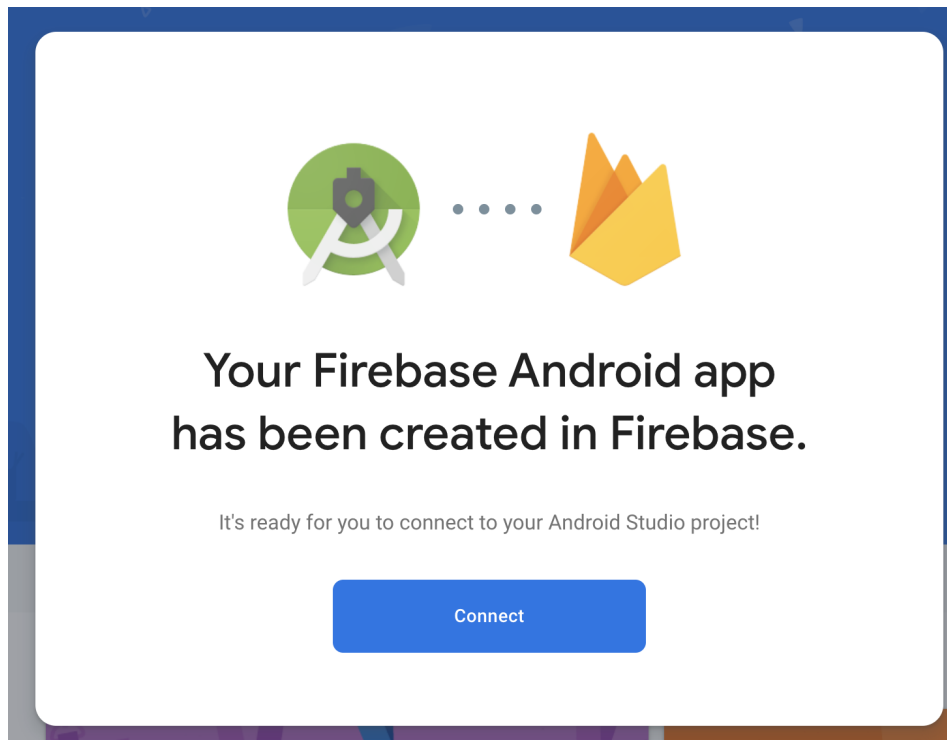
El punto 1 y 2 son los que nos importan por ahora



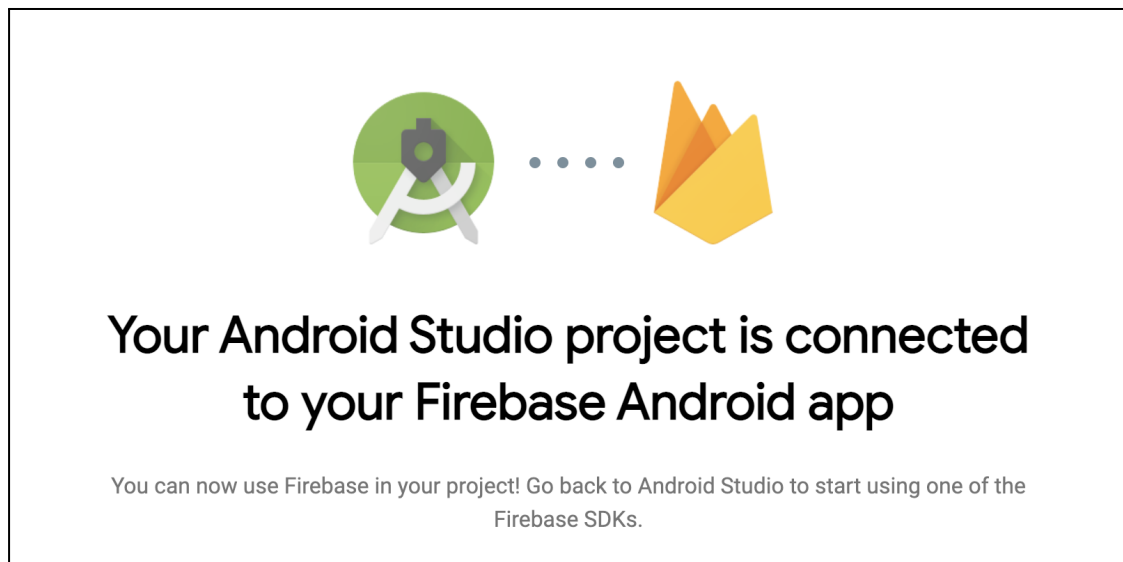
Cuando selecciones “Connect to Firebase” se abrirá tu navegador con la siguiente página, selecciona el proyecto que creaste en Firebase:



Selecciona "Connect"



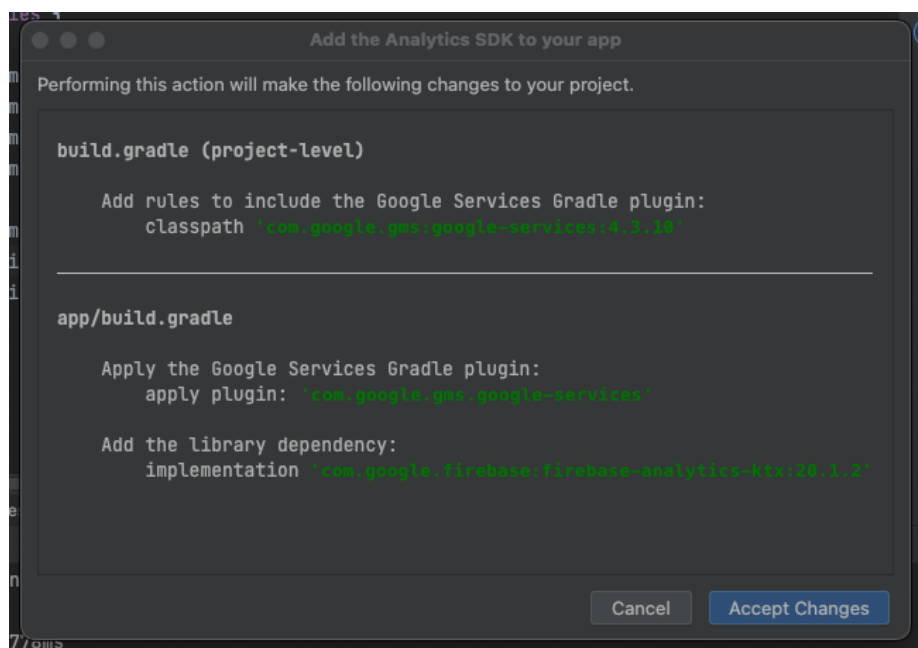
Una vez terminado verás el siguiente mensaje:



Y en Android Studio verás la siguiente notificación:



Luego presiona "Add Analytics SDK to your app", esto aplicará los siguientes cambios en tu proyecto:



Ahora repite el mismo proceso, esta vez seleccionando Crashlytics.

Al finalizar verás que tu build.gradle habrá cambiado, agregando las siguientes dependencias:

```
implementation 'com.google.firebase:firebase-analytics-ktx:20.1.2'  
implementation 'com.google.firebase:firebase-crashlytics:18.2.9'  
implementation 'com.google.firebase:firebase-analytics:20.1.2'
```

Necesitamos intencionalmente generar un crash, esto lo podemos hacer de forma muy sencilla tirando una excepción.

En el MainActivity de tu proyecto, crea un botón, el cual, al presionarlo, llame el siguiente código:

```
implementation 'com.google.firebase:firebase-analytics-ktx:21.2.1'  
implementation 'com.google.firebase:firebase-crashlytics:18.3.6'  
implementation 'com.google.firebase:firebase-crashlytics-ktx:18.3.6'  
implementation 'com.google.firebase:firebase-analytics:21.2.1'
```

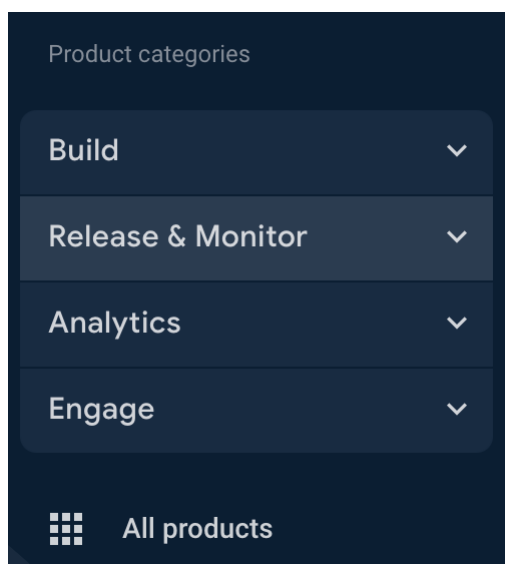
En el MainActivity de tu proyecto, crea un botón, el cual, al presionarlo, llame el siguiente código:

```
binding.btnCrash.setOnClickListener {  
    throw RuntimeException("Test Crash")  
}
```

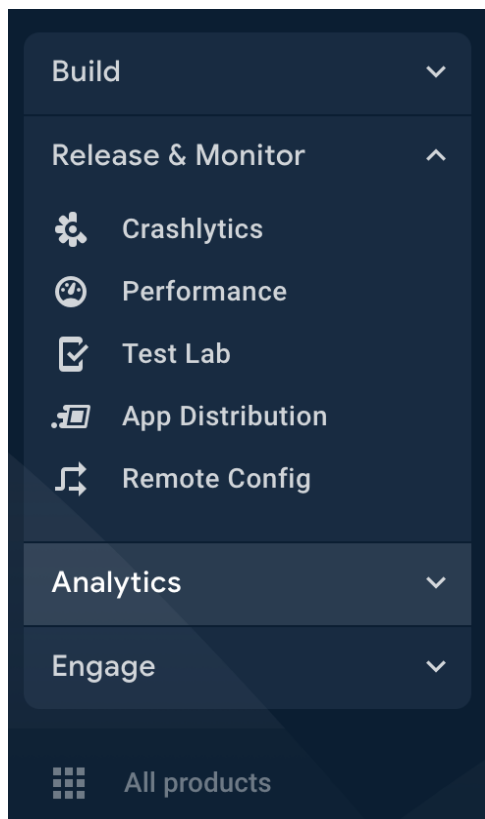
Lo que estamos haciendo es simplemente generando una excepción, la que hará que la aplicación se cierre y envíe la notificación de crash a Crashlytics.

Después de correr la aplicación y presionar el botón de crash, podemos ir a Crashlytics para ver el reporte de crash. Para esto, anda a la consola de Firebase y selecciona tu proyecto: <https://console.firebase.google.com>

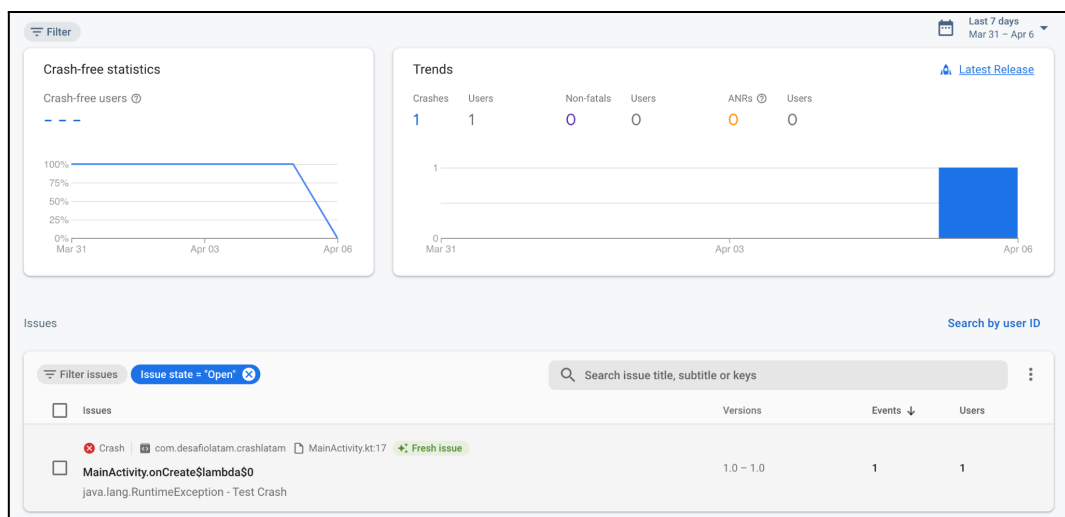
En el menú de la izquierda podrás ver algo similar a esto:



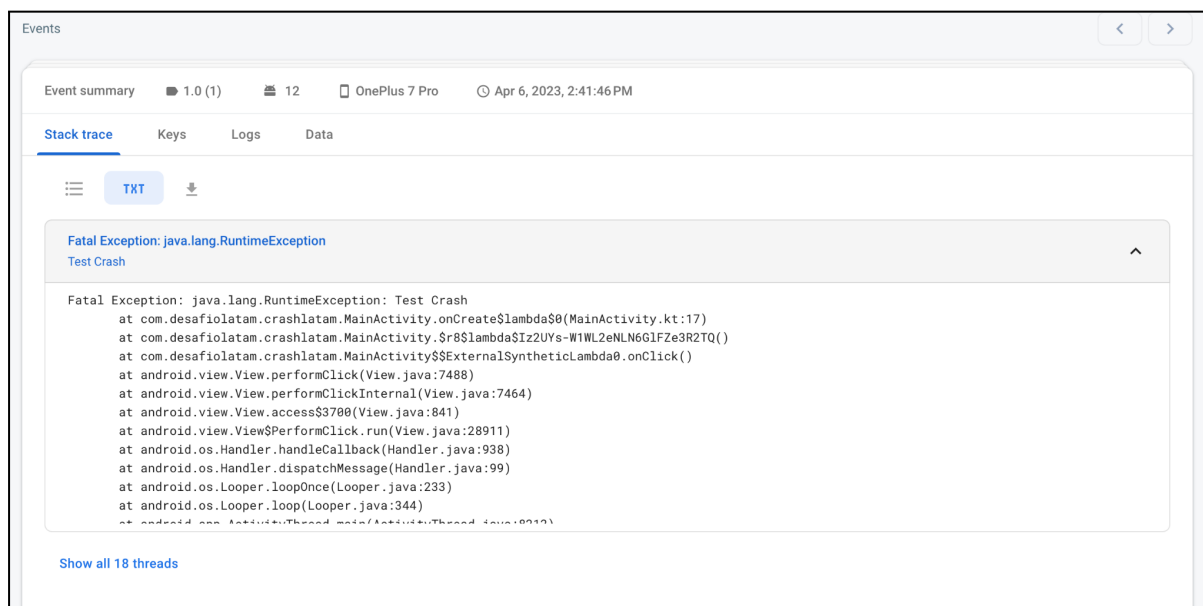
Expande la opción "Release & Monitor" y verás Crashlytics, selecciona y podrás ver el primer reporte de fallo generado por tu app.



Primero verás un listado de crashes, en este caso deberíamos tener solamente uno:



Y al abrir el detalle, deberíamos ver algo similar a lo siguiente:



Algo extra que podemos hacer es enviar un log o keys personalizados, lo podemos hacer de la siguiente forma:

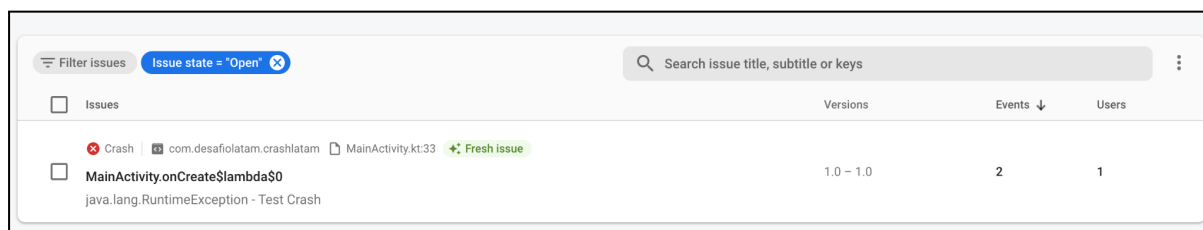
Antes de este bloque de código:

```
binding.btnCrash.setOnClickListener {  
    throw RuntimeException("Test Crash")  
}
```

Definimos el mensaje del log y una key:

```
Firebase.crashlytics.log("Este es un mensaje de log")  
  
Firebase.crashlytics.setCustomKeys {  
    key("string_key", "Hola crash!!!")  
}
```

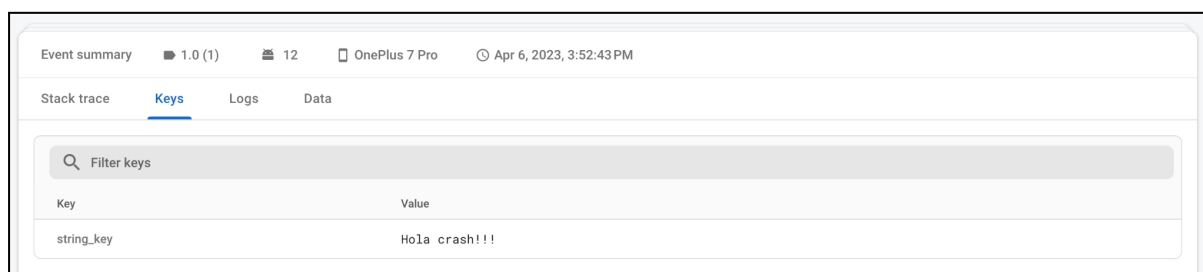
Al ver el dashboard de Crashlytics podemos ver que hay un nuevo crash, ya que usamos el ejemplo anterior, Crashlytics agrupa este tipo de eventos.



Issues	Versions	Events ↓	Users
<div>Crash</div> <div>com.desafiolatam.crashlatam MainActivity.kt:33</div> <div>MainActivity.onCreate\$lambda\$0</div> <div>java.lang.RuntimeException - Test Crash</div>	1.0 - 1.0	2	1

Ahora podemos ver que la cantidad de eventos aumentó a 2. Sin embargo, el crash sigue ocurriendo en el mismo equipo, es decir es el mismo usuario, si vemos el detalle y vemos dentro de Keys y Logs encontraremos más información.

Para el caso de Keys:



Key	Value
string_key	Hola crash!!!

Y en el caso de Logs:



#	Time (newest first)	Source	Log
1	03:52:37.268 PM	screen_view	screen_view
2	03:52:37.245 PM	session_start	session_start
3	03:52:37.178 PM		Este es un mensaje de log



¡Con esto hemos terminado!

La idea de usar Crashlytics es poder capturar información del momento en el que ocurre algo inesperado o no controlado como es un crash, también se puede utilizar para ver ANRs (Application Not Responding).



¡Manos a la obra! - Nombre de la actividad

Usando la opción de logs y keys personalizados, envía a Crashlytics lo siguiente:

- **Logs**

"Este es un mensaje enviado con: "Firebase.crashlytics.log" "

- **Keys:**

Key	Value
string_key	Hola crash!!!
boolean_key	true
double_key	1.0
float_key	1.0f
int_key	1

Respuesta:

El alumno deberá “escapar” las comillas en el mensaje para que se muestre de forma adecuada

```
Firestore.crashlytics.log("Este es un mensaje enviado con:  
\"Firestore.crashlytics.log\" ")
```

```
Firestore.crashlytics.setCustomKeys {  
    key("string_key", "Hola crash!!!")  
    key("boolean_key", true)  
    key("double_key", 1.0)  
    key("float_key", 1.0f)  
    key("int_key", 1)  
}  
  
binding.btnCrash.setOnClickListener {  
    throw RuntimeException("Test Crash")  
}
```


Preguntas de proceso

Reflexiona:

- ¿Habías utilizado Firebase anteriormente? ¿Qué te pareció su uso en estos ejercicios? ¿Los hizo más sencillos o complejizó tu comprensión de ello?
- ¿Existe algún contenido que aún no puedas comprender cabalmente? ¿Qué es lo que crees que lo hace más complejo? ¿Se te ocurre alguna manera de poder hacer más sencillo ese aprendizaje?



Preguntas de cierre

- ¿Cuál es la diferencia entre un crash y un ANR?
- ¿Puede Crashlytics capturar ANR, necesito alguna configuración extra?
- ¿Cómo sé donde puede ocurrir un crash para capturarlo?
- ¿Hay alguna forma de generar un crash y evitar que se cierre la aplicación?

Referencias bibliográficas

- Android ANRs:
<https://developer.android.com/topic/performance/vitals/anr>
- Documentación Crashlytics:
- <https://firebase.google.com/docs/crashlytics?hl=en&authuser=0>