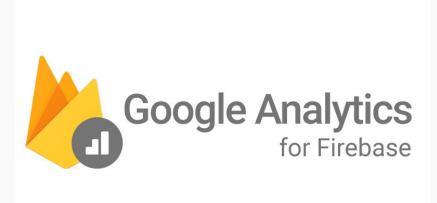
Universidad Nacional de La Matanza

Google Analytics for Firebase

Intro

Herramienta de medición que proporciona estadísticas sobre el uso de las apps y la participación de los usuarios.

Permite entender cómo se comportan los usuarios y así tomar decisiones sobre cambios o mejoras en nuestra app



Funcionalidades

Registro de eventos

Definir propiedades y segmentos de usuarios

Generación de informes

Seguimiento de pantallas

Integración con otras herramientas

Crashlytics

Firebase remote config

[otras más, como Google Tag Manager, Big Query; pero a estas no le daremos foco en la cátedra]

Configuración

Habilitar en Firebase

Al crear un projecto nuevo, tildar la opción de Analytics

Para verificar o activar en un proyecto ya creado

Página de configuración -> Integraciones -> Google Analytics

Configuración

En el archivo build.gradle, a nivel de app

```
dependencies {
    // Import the BoM for the Firebase platform
    // BoM permite que la app siempre utilice las versiones compatibles de las bibliotecas
    implementation platform('com.google.firebase:firebase-bom:27.1.0')

// Declarar la dependencia de Analytics CON BoM
    implementation 'com.google.firebase:firebase-analytics-ktx'

// Declarar la dependencia de Analytics SIN BoM
    // sin BoM, se debe especificar la versión de la biblioteca importada
    implementation 'com.google.firebase:firebase-analytics-ktx:18.0.3'
```

Configuración

En la actividad donde aplicaremos Analytics

```
Agregar el import:

com.google.firebase.analytics.FirebaseAnalytics

Declarar el objeto:

private lateinit var firebaseAnalytics: FirebaseAnalytics

Finalmente, en el método OnCreate():

firebaseAnalytics = Firebase.analytics
```

Los eventos proporcionan información sobre lo que sucede en nuestra app (acciones del usuario, eventos del sistema, errores)

Algunos eventos automáticos:

- screen_view
- session_start
- first_open

- app_update
- app_exception
- error

Método <u>logEvent()</u>

```
firebaseAnalytics.logEvent(FirebaseAnalytics.Event.SELECT_ITEM) {
    param(FirebaseAnalytics.Param.ITEM_ID, id)
    param(FirebaseAnalytics.Param.ITEM_NAME, name)
    param(FirebaseAnalytics.Param.CONTENT_TYPE, "image")
}
```

Eventos personalizados

```
firebaseAnalytics.logEvent("Movimiento") {
   param("usuario", jugador)
   param("position", position)
   param("partida", partida)
}
```

Parámetros predeterminados

```
val parameters = Bundle().apply {
    this.putString("level_name", "Caverns01")
    this.putInt("position", 0)
}
firebaseAnalytics.setDefaultEventParameters(parameters)
```

Propiedades de usuarios

Propiedades de usuario

Atributos que definimos para describir segmentos de usuarios.

- Edad
- Género
- Interés
- País
- Marca del dispositivo
- Modelo del dispositivo

- Tienda de apps
- Versión de la app
- Versión de SO
- Plataforma
- Idioma
- Nuevo o establecido

Propiedades de usuario

Analytics permite crear hasta 25 propiedades personalizadas.

Para crear una propiedad:

Ir a la Consola de Firebase, en la sección Analytics, menú Custom Definitions

Para configurar una propiedad en el código:

```
// Tipo de membresía: normal, premium
firebaseAnalytics.setUserProperty("membership", membresia)
```

Seguimiento de pantallas

Seguimiento de pantallas

Automático en cada Activity, guardado con el evento screen_view

Si hay un Activity por pantalla, realiza el seguimiento de transición de las mismas

Seguimiento de pantallas

Seguimiento manual

```
firebaseAnalytics.logEvent(FirebaseAnalytics.Event.SCREEN_VIEW) {
    param(FirebaseAnalytics.Param.SCREEN_NAME, screenName)
    param(FirebaseAnalytics.Param.SCREEN_CLASS, "MainActivity")
}
```

Bibliografía

- <u>Documentación oficial de Google Analytics for Firebase</u>
- Eventos predefinidos
- Parámetros predefinidos

¿Preguntas?