

# Medir, medir y medir

Formas de conocer a nuestros usuarios



# Ciclo de vida de una App

¿Cómo se usa  
nuestra  
aplicación?



1

Análisis

2

Desarrollo

3

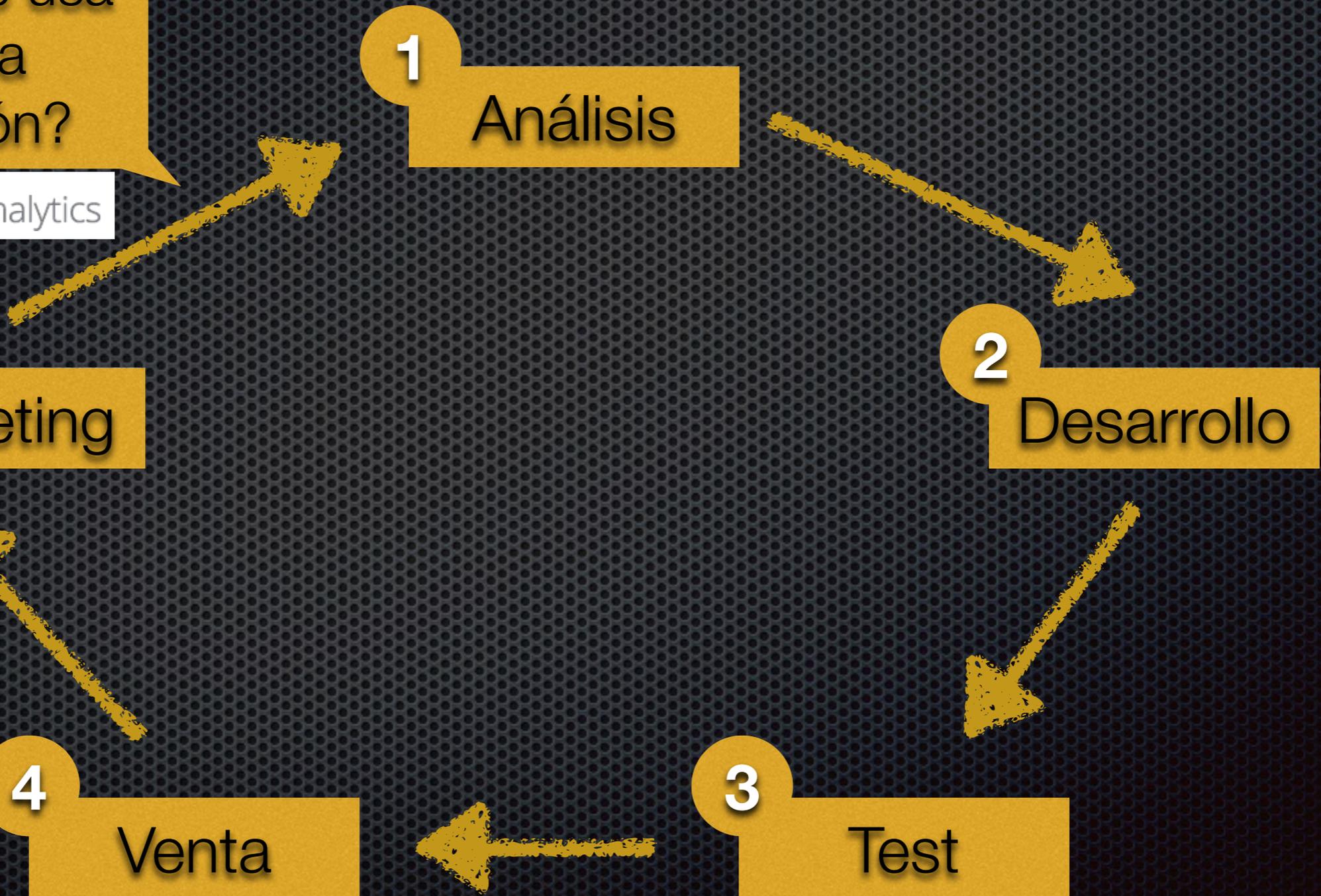
Test

4

Venta

5

Marketing



# ¿Qué podemos medir?

- ✦ Medir Pantallas
- ✦ Medir Eventos
- ✦ Medir Campañas
- ✦ Medir Compras dentro de la App
- ✦ Medir los tiempos de carga ...



¿Por dónde  
comenzamos a medir?

# ¿Cómo medimos?

- Solicitar un identificador para nuestra App
- Integrar Google Analytics SDK en nuestro proyecto
  1. Actualizar el AndroidManifest.xml
  2. Inicializar el tracker
  3. Enviando información
  4. Configuración
  5. Opciones avanzadas

# 1. Actualizar el AndroidManifest.xml

- ✦ Añadir los siguientes permisos:

```
<uses-permission android:name="android.permission.INTERNET" />
```

```
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

## 2. Inicializar el tracker

- ✦ Cada programador debe gestionar el tracker por si mismo
- ✦ Se recomienda crearlo y gestionarlo en la clase *Application* añadiendo una variable *Tracker* a la misma:

```
Tracker tracker;
```

## 2. Inicializar el tracker

- Añadir el siguiente método a la clase *Application*:

```
synchronized Tracker getTracker() {  
    if (!tracker) {  
        GoogleAnalytics analytics = GoogleAnalytics.getInstance(this);  
        tracker = analytics.newTracker(PROPERTY_ID);  
    }  
    return tracker;  
}
```

# 3. Enviando información

- Para enviar información añadiremos lo siguiente:

```
// Obtenemos el tracker de la aplicación
Tracker t = ((AnalyticsSampleApp)
getActivity().getApplication()).getTracker();

// Asignamos el nombre de la pantalla actual.
t.setScreenName(nom_Pantalla);

// Enviamos la información.
t.send(new HitBuilders.AppViewBuilder().build());
```

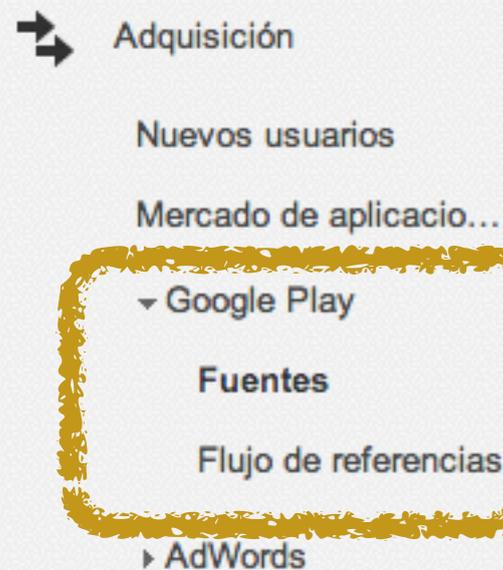
# 3. Enviando información

- ✦ Para enviar información de un evento:

```
// Enviamos la información de un evento.  
t.send(new HitBuilders.EventBuilder()  
    .setCategory(getString(categoryId))  
    .setAction(getString(actionId))  
    .setLabel(getString(labelId))  
    .build());
```

# 4. Configuración

- ✦ **Existe un apartado específico para Google Play que se deberá configurar para que funcione:**



## **Este informe solicita que enlace sus aplicaciones con Google Play**

Para que los datos aparezcan en este informe, es necesario que enlace Google Analytics con su Consola de Google Play para programadores.

### **¿Para qué hay que enlazar Google Play con Google Analytics?**

Al enlazar Google Analytics con Google Play, tendrá acceso a informes que le ayudarán a medir el interés por su aplicación. También sabrá cómo encuentran su aplicación los usuarios de Google Play y podrá consultar el número de veces que se ha visto la página de descripción de la aplicación, la cantidad de descargas y cuántas veces se ha iniciado por primera vez. [Más información](#)

[Configurar el enlace de aplicaciones](#)

# 5. Opciones avanzadas

- **Dry Run:** Cuando se activa no envía la información a Google Analytics. Sirve mientras estamos en desarrollo y no queremos que la información se integre con la de producción.

// Activamos el modo de desarrollo.

```
GoogleAnalytics.getInstance(this).setDryRun(true);
```

# 5. Opciones avanzadas

- ✦ **App-level Opt Out:** Permite detener toda la recogida de datos que realiza la aplicación.

// Detenemos toda la recogida de información.

```
GoogleAnalytics.getInstance(this).setAppOptOut(true);
```

# ¿Qué podemos obtener?

- ✦ **Tiempo real**
- ✦ **Público:** Idioma, ubicación, versión, dispositivo, red, sistema operativo ...
- ✦ **Adquisición:** Google play, AdWords ...
- ✦ **Comportamiento:** tiempo por pantalla, eventos más frecuentes, eventos que no se usan ...
- ✦ **Conversiones:** logros a contabilizar

# Tiempo real

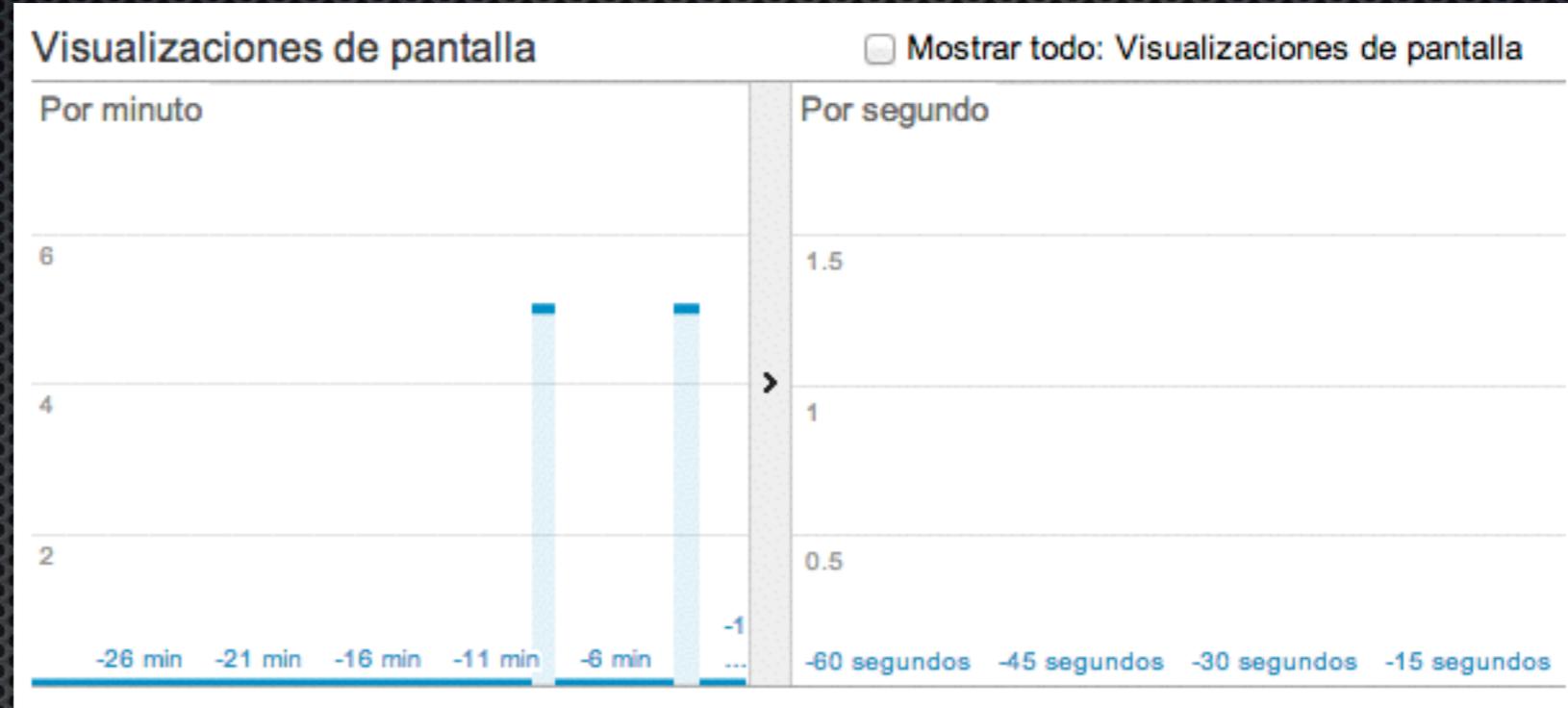
**En este momento**

# 1

usuarios activos en la aplicación

■ MÓVIL

100%



**Principales versiones de las aplicaciones:**

|    | Versión de la aplicación | Usuarios activos ↓ |
|----|--------------------------|--------------------|
| 1. | 1.0                      | 1                  |

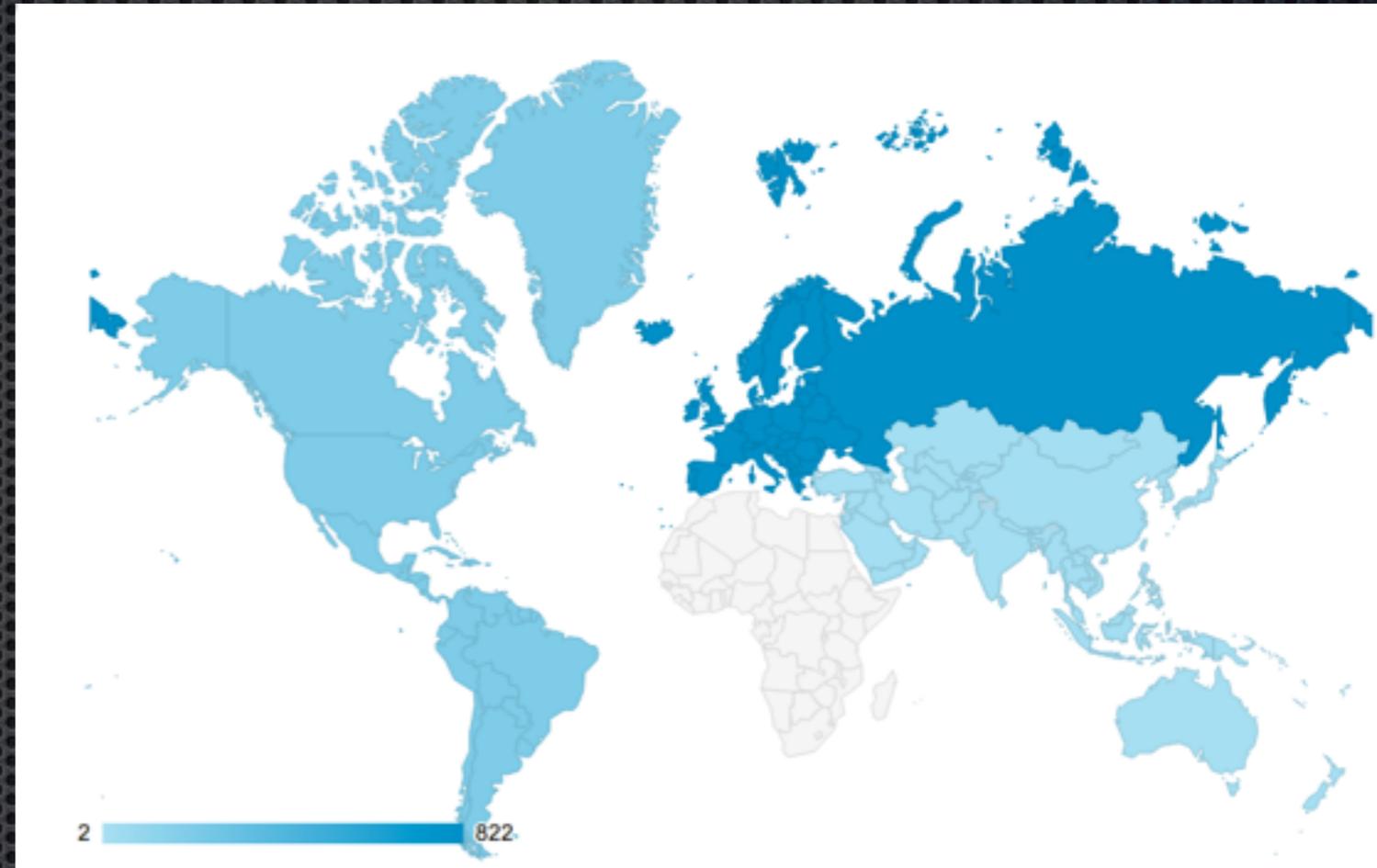
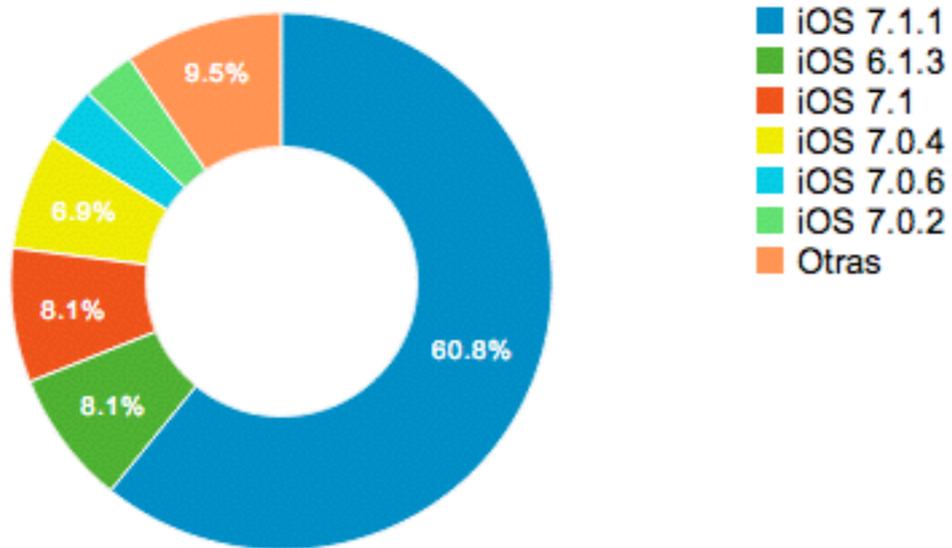
**Principales pantallas activas:**

|    | Nombre de pantalla activa | Usuarios activos ↓ |
|----|---------------------------|--------------------|
| 1. | EJERCICIOS                | 1 100,00%          |



# Público: idioma, ubicación...

## Versiones del SO



| Idioma     | Sesiones | % Sesiones |
|------------|----------|------------|
| 1. es      | 881      | 83,11%     |
| 2. en      | 113      | 10,66%     |
| 3. fr      | 36       | 3,40%      |
| 4. ca      | 10       | 0,94%      |
| 5. en-gb   | 7        | 0,66%      |
| 6. de      | 6        | 0,57%      |
| 7. zh-hans | 6        | 0,57%      |
| 8. es-mx   | 1        | 0,09%      |

[ver todo el informe](#)

## Modelos de dispositivos principales



Apple  
iPad

68,73%



Apple  
iPhone

30,42%



Apple  
iPod Touch

0,85%

# Caso práctico Mathcubes



SchoolCubes



# Comportamiento



# Consecución de objetivos

- ✦ Nos permite definir lo que es para nosotros la consecución de un objetivo y ver gráficos sobre las consecuciones de los mismos.



# Más información

- ✦ <https://developers.google.com/analytics/devguides/collection/android/v4/?hl=e>
- ✦ [https://www.youtube.com/watch?v=p37g9h\\_Zkw0](https://www.youtube.com/watch?v=p37g9h_Zkw0)
- ✦ <http://www.google.com/analytics/>