



**UNIVERSIDAD  
DE ANTIOQUIA**  
1803

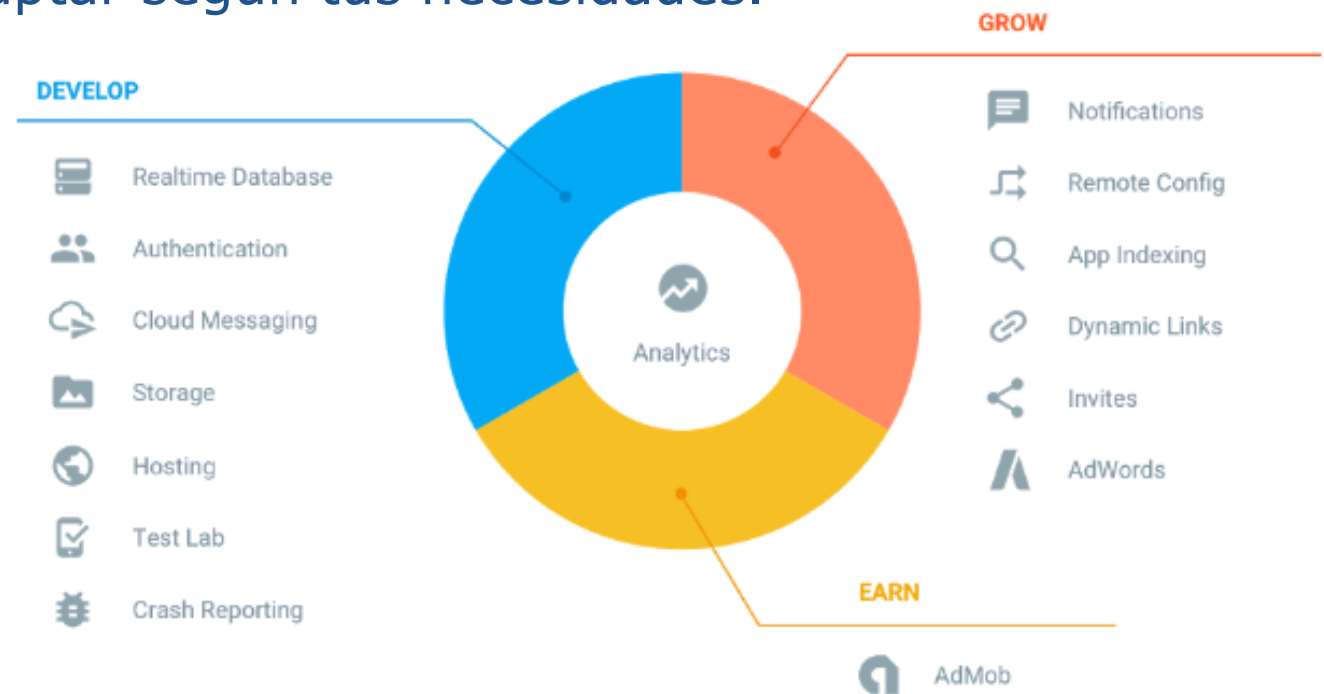
## **Sesión 9. Firebase**

Ing. Edwin Andrés Cubillos Vega Msc.





## ¿Qué es Firebase?

Firebase es una plataforma móvil que te permite **desarrollar** rápidamente apps de alta calidad, **aumentar** tu base de usuarios y **ganar** más dinero. Firebase contiene funciones complementarias que puedes combinar y adaptar según tus necesidades.



← → ↻ <https://firebase.google.com> ☆ ABP

 **Firebase** Home Features Pricing Docs Customers Support


🔍 Search [Go to console](#) 

## App success made simple

The tools and infrastructure you need to build better apps and grow successful businesses

[GET STARTED FOR FREE](#)

[Attend](#) or [live-stream](#) the Firebase Dev Summit in Berlin

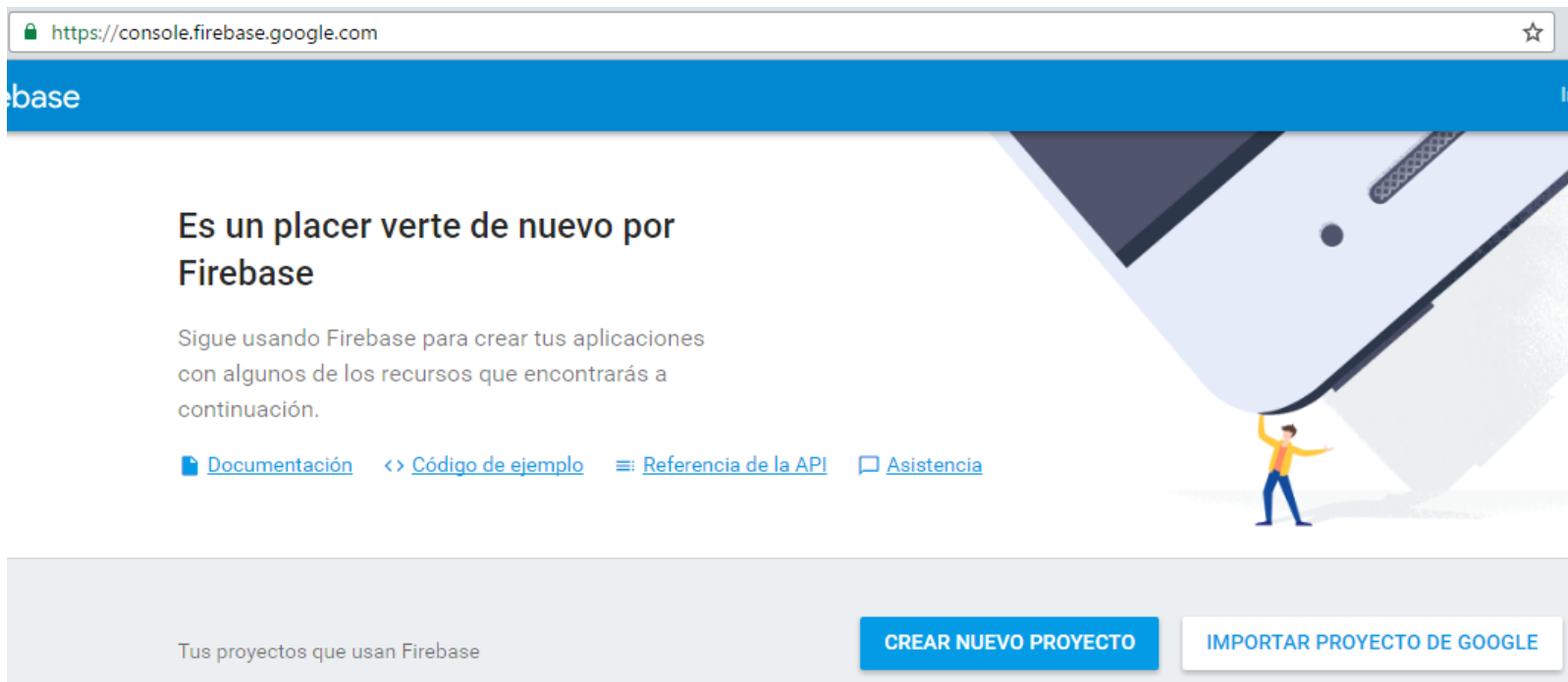


## Move fast

Firebase is a mobile platform that helps you quickly **develop** high-quality apps, **grow** your user base, and **earn** more money. Firebase is



## ❖ Crear nuevo Proyecto



The screenshot shows the Firebase console homepage. At the top, there's a blue header with the word "firebase" on the left and "In" on the right. Below the header, the main content area has a large illustration of a person in a yellow shirt and blue pants standing next to a large, stylized, light blue and grey geometric shape that resembles a computer monitor or a piece of hardware. The text "Es un placer verte de nuevo por Firebase" is prominently displayed. Below this, a paragraph says "Sigue usando Firebase para crear tus aplicaciones con algunos de los recursos que encontrarás a continuación." There are four links: "Documentación", "> Código de ejemplo", "Referencia de la API", and "Asistencia". At the bottom, there's a light grey bar with the text "Tus proyectos que usan Firebase" on the left and two buttons: "CREAR NUEVO PROYECTO" (blue) and "IMPORTAR PROYECTO DE GOOGLE" (white with blue border).

https://console.firebase.google.com

base

### Es un placer verte de nuevo por Firebase

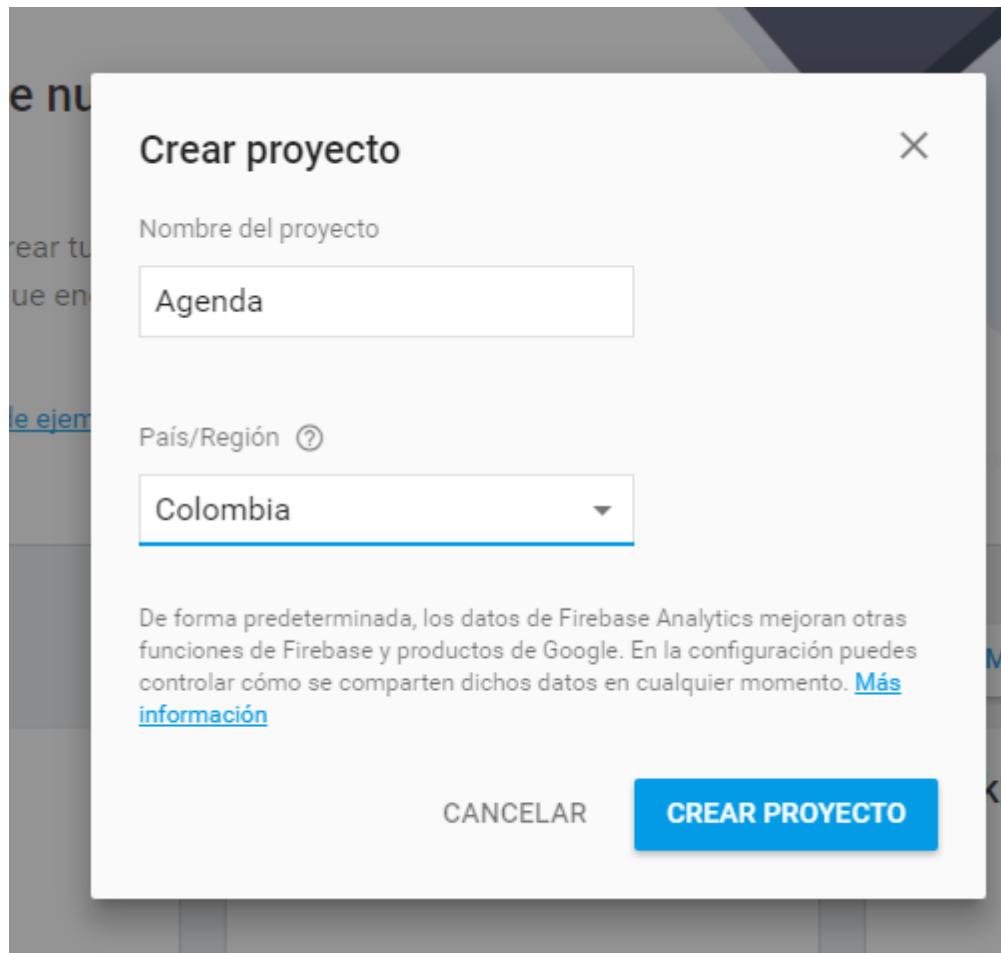
Sigue usando Firebase para crear tus aplicaciones con algunos de los recursos que encontrarás a continuación.

[Documentación](#) [> Código de ejemplo](#) [Referencia de la API](#) [Asistencia](#)

Tus proyectos que usan Firebase

[CREAR NUEVO PROYECTO](#) [IMPORTAR PROYECTO DE GOOGLE](#)

## ❖ Configurar nombre y país/región



**Crear proyecto** ✕

Nombre del proyecto

Agenda

País/Región ⓘ

Colombia ▼

De forma predeterminada, los datos de Firebase Analytics mejoran otras funciones de Firebase y productos de Google. En la configuración puedes controlar cómo se comparten dichos datos en cualquier momento. [Más información](#)

CANCELAR CREAR PROYECTO

## ❖ Seleccionar Sistema Operativo Android

The screenshot displays the Firebase console's 'Overview' page. At the top, a blue header bar contains the Firebase logo on the left and links for 'Agenda' and 'Ir a la documentación' on the right. Below this, a left sidebar lists various services under 'DEVELOP' (Analytics, Authentication, Database, Storage, Hosting, Test Lab, Crash Reporting) and 'GROW' (Notifications, Remote Config). The main content area, titled 'Overview', welcomes the user and presents three large circular buttons: 'iOS', 'Android', and 'Web'. The 'Android' button, featuring the Android robot icon, is circled in green. Below each button is the text 'Añade Firebase a tu aplicación de [platform]'. At the bottom of the console, there's a 'Descubre Firebase' section with colorful illustrations.

- ❖ Se configura el nombre del paquete, el apodo de la aplicación y el certificado SHA-1

**Añade Firebase a tu aplicación de Android** ✕

1 — 2 — 3

Introduce los detalles de la aplicación   Copiar archivo de configuración   Añadir a build.gradle

Nombre del paquete ⓘ

com.yourapp.android

Apodo de la aplicación (opcional) ⓘ

Freemium Android App

Certificado de firma de depuración SHA-1 (opcional) ⓘ

00:00

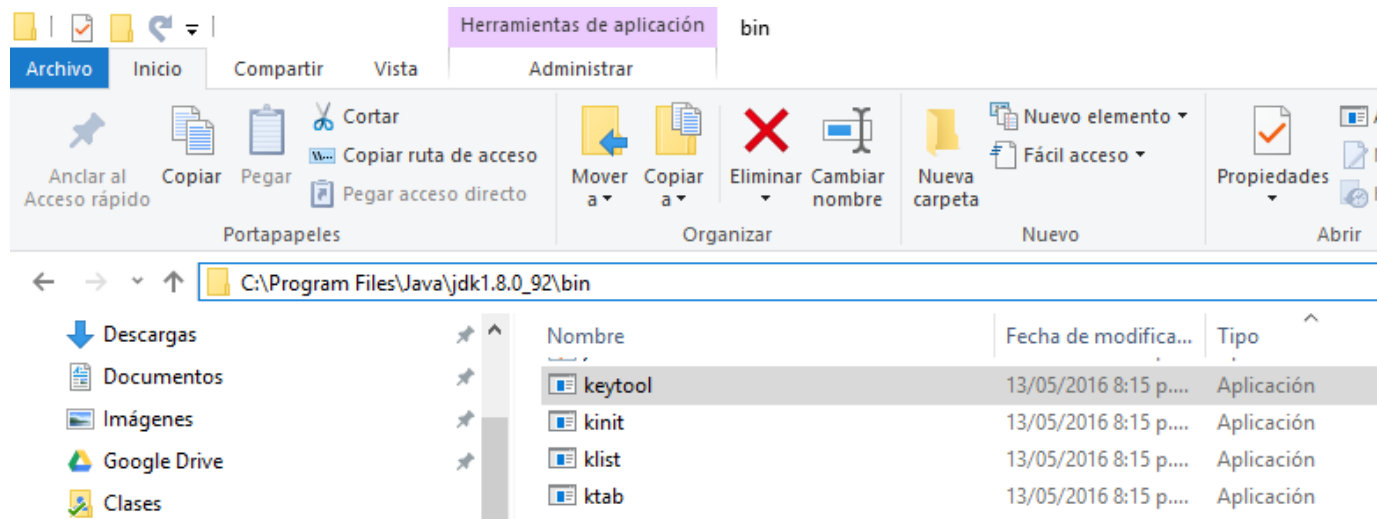
Obligatorio para enlaces dinámicos, invitaciones y compatibilidad con inicio de sesión de Google en Auth. Edita SHA-1 en la configuración.

CANCELAR   AÑADIR APLICACIÓN

[descarga google-services.json para tu aplicación](#)



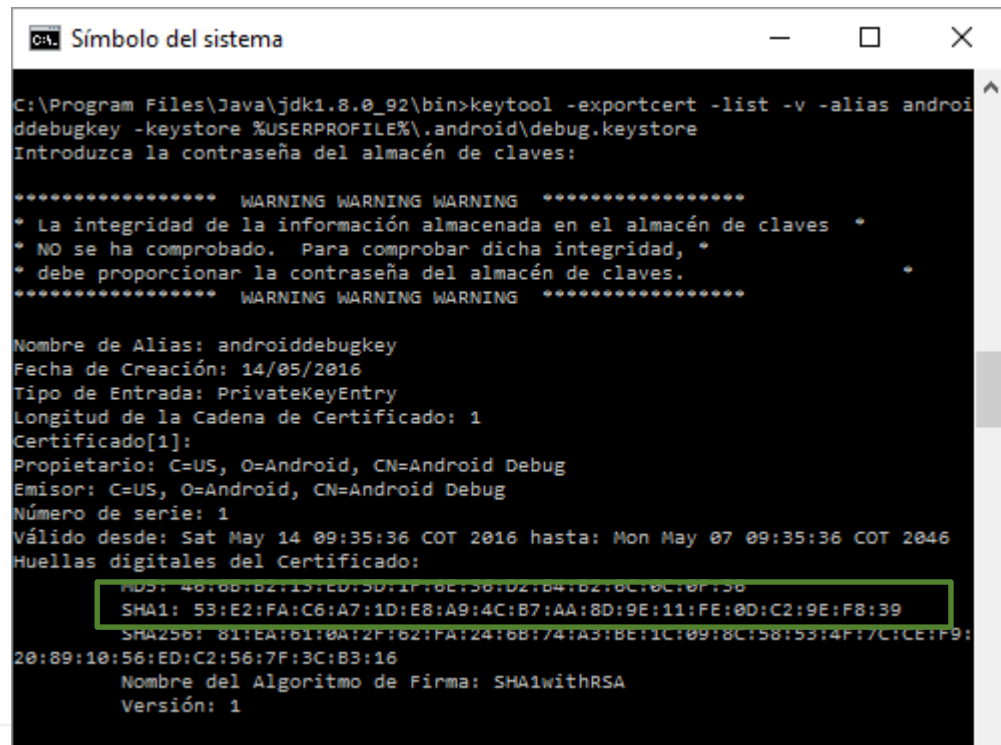
- ❖ Para generar el certificado SHA se utiliza la herramienta keytool



- ❖ Se abre el símbolo del sistema y se va a la ruta donde está keytool y se copia y se pega el comando



Se abre el símbolo del sistema y se va a la ruta donde esta keytool y se copia y se pega el comando



```
C:\Program Files\Java\jdk1.8.0_92\bin>keytool -exportcert -list -v -alias androiddebugkey -keystore %USERPROFILE%\.android\debug.keystore
Introduzca la contraseña del almacén de claves:

***** WARNING WARNING WARNING *****
* La integridad de la información almacenada en el almacén de claves *
* NO se ha comprobado. Para comprobar dicha integridad, *
* debe proporcionar la contraseña del almacén de claves. *
***** WARNING WARNING WARNING *****

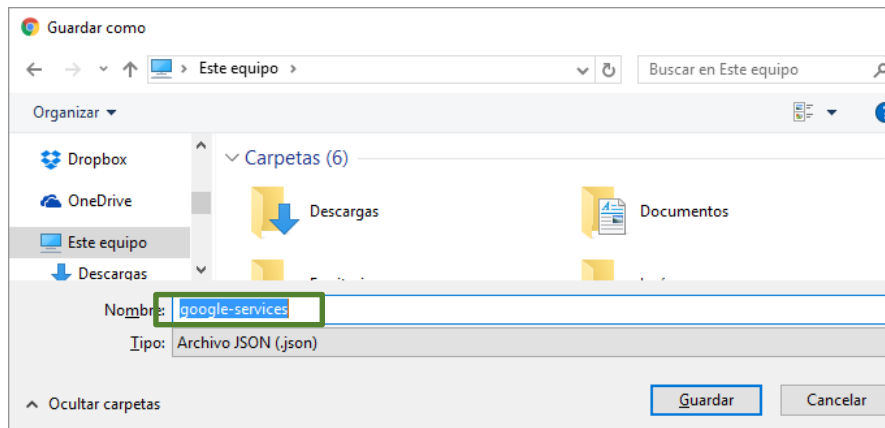
Nombre de Aliás: androiddebugkey
Fecha de Creación: 14/05/2016
Tipo de Entrada: PrivateKeyEntry
Longitud de la Cadena de Certificado: 1
Certificado[1]:
Propietario: C=US, O=Android, CN=Android Debug
Emisor: C=US, O=Android, CN=Android Debug
Número de serie: 1
Válido desde: Sat May 14 09:35:36 COT 2016 hasta: Mon May 07 09:35:36 COT 2046
Huellas digitales del Certificado:
MD5: 46:6B:62:13:ED:3D:1F:8E:58:D2:64:B2:6C:0C:0F:36
SHA1: 53:E2:FA:C6:A7:1D:E8:A9:4C:B7:AA:8D:9E:11:FE:0D:C2:9E:F8:39
SHA256: 81:EA:61:0A:2F:62:F4:24:6B:74:A3:BE:1C:09:8C:58:53:4F:7C:CE:F9:
20:89:10:56:ED:C2:56:7F:3C:B3:16
Nombre del Algoritmo de Firma: SHA1withRSA
Versión: 1
```

MAC/LINUX

WINDOWS

```
keytool -exportcert -list -v \
-alias androiddebugkey -keystore %USERPROFILE%\.android\debug.keystore
```

- ❖ Firestore entregará un archivo de configuración, almacenarlo en la ruta solicitada

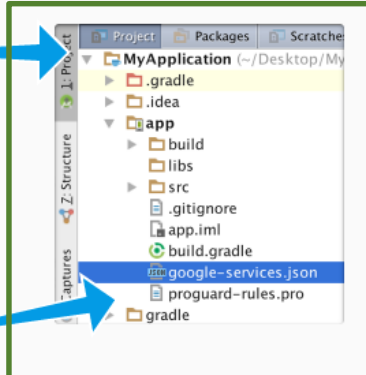


### Añade Firestore a tu aplicación de Android

- 1 Introduce los detalles de la aplicación
- 2 Copiar archivo de configuración
- 3 Añadir a build.gradle

Cambia a la vista **Proyecto** de Android Studio para ver el directorio "root" de tu proyecto.

Mueve el archivo `google-services.json` que acabas de descargar al directorio "root" del módulo de tu aplicación de Android.



`google-services.json`

¿Ya has añadido las dependencias?  
[Saltar a la consola](#)

**CONTINUAR**

- ❖ Ahora debemos configurar en el gradle de la aplicación:

1. build.gradle de proyecto (<project>/build.gradle):

```
buildscript {  
    dependencies {  
        // Add this line  
        classpath 'com.google.gms:google-services:3.0.0'  
    }  
}
```

- ❖ Y en el gradle del modulo

2. build.gradle de aplicación (<project>/<app-module>/build.gradle):

```
...  
// Add to the bottom of the file  
apply plugin: 'com.google.gms.google-services'
```

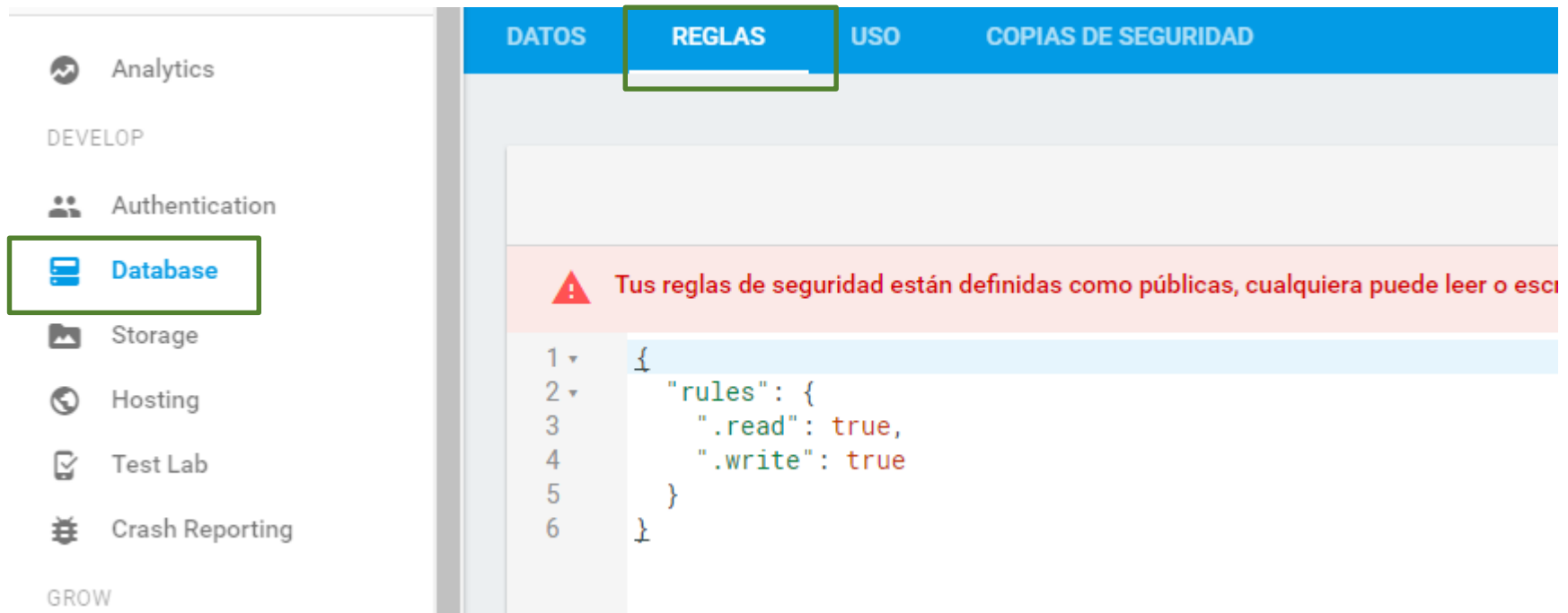
## ❖ En el gradle del módulo agregamos

```
android {  
    compileSdkVersion 24  
    buildToolsVersion "24.0.3"  
  
    packagingOptions{  
        exclude 'META-INF/LICENSE'  
        exclude 'META-INF/LICENSE-FIREBASE.txt'  
        exclude 'META-INF/NOTICE'  
    }  
}
```

## ❖ Y en las dependencias, luego se sincroniza el gradle

```
dependencies {  
    compile fileTree(dir: 'libs', include: ['*.jar'])  
    androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {  
        exclude group: 'com.android.support', module: 'support-annotations'  
    })  
    compile 'com.android.support:appcompat-v7:24.2.1'  
    compile 'com.firebase:firebase-client-android:2.5.2+'
```

- ❖ Cambiar las reglas de seguridad de la base de datos para que sea pública y se pueda acceder sin login



The screenshot shows the Firebase console interface. On the left sidebar, the 'Database' option is highlighted with a green box. The main panel has a blue header with tabs: 'DATOS', 'REGLAS' (highlighted with a green box), 'USO', and 'COPIAS DE SEGURIDAD'. Below the tabs, a red warning banner states: '¡ Tus reglas de seguridad están definidas como públicas, cualquiera puede leer o escribir tus datos. ¡ ¡ ¡'. Below the warning, a code editor shows the following JSON rules:

```
1 {  
2   "rules": {  
3     ".read": true,  
4     ".write": true  
5   }  
6 }
```

- ❖ La aplicación a desarrollar es una agenda simple



En Android Studio se crea la clase Contacto.java

```
public class Contacto {

    private String nombre, telefono, mail, id;

    public Contacto() {
    }

    public Contacto(String nombre, String telefono, String mail, String id) {
        this.nombre = nombre;
        this.telefono = telefono;
        this.mail = mail;
        this.id = id;
    }

    public String getNombre() { return nombre; }

    public String getTelefono() { return telefono; }

    public String getMail() { return mail; }

    public String getId() { return id; }
}
```



## Configuramos lo necesario en el MainActivity

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {

    EditText eId, eNombre, eTelefono, eCorreo;

    String nombre, correo, telefono, codigo;
    Button bInsertar, bActualizar, bBorrar, bBuscar, bLimpiar;
    Integer id=0;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        eId = (EditText) findViewById(R.id.eId);
        eNombre = (EditText) findViewById(R.id.eNombre);
        eTelefono = (EditText) findViewById(R.id.eTelefono);
        eCorreo = (EditText) findViewById(R.id.eMail);
        bInsertar = (Button) findViewById(R.id.bInsertar);
        bActualizar = (Button) findViewById(R.id.bActualizar);
        bBorrar = (Button) findViewById(R.id.bBorrar);
        bBuscar = (Button) findViewById(R.id.bBuscar);
        bLimpiar = (Button) findViewById(R.id.bLimpiar);

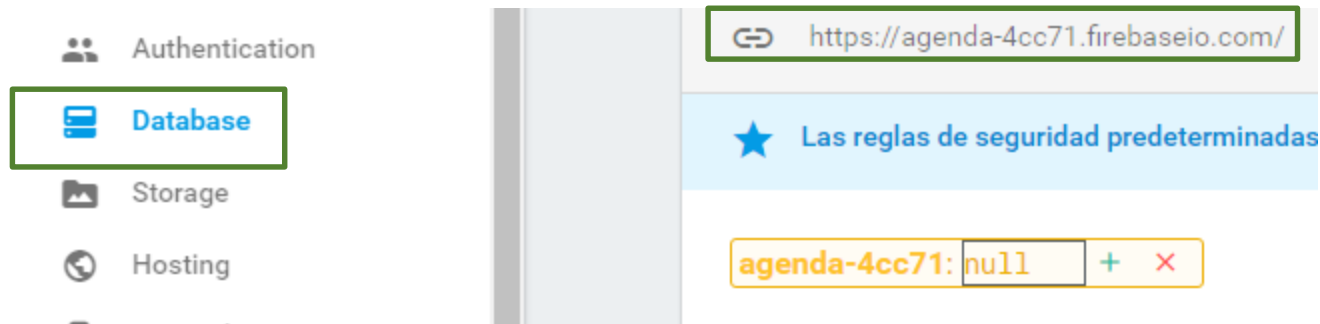
        bInsertar.setOnClickListener(this);
        bActualizar.setOnClickListener(this);
        bBorrar.setOnClickListener(this);
        bBuscar.setOnClickListener(this);
        bLimpiar.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {
        nombre = eNombre.getText().toString();
        correo = eCorreo.getText().toString();
        telefono = eTelefono.getText().toString();
        codigo = eId.getText().toString();

        switch (v.getId()) {
            case R.id.bLimpiar:
                eNombre.setText("");
                eTelefono.setText("");
                eCorreo.setText("");
                eId.setText("");
                break;
            case R.id.bInsertar:
                break;
            case R.id.bActualizar:
                break;
            case R.id.bBorrar:
                break;
            case R.id.bBuscar:
                break;
        }
    }
}
```

- ❖ Se crea un string con los datos de la URL de la base de datos

```
private String FIREBASE_URL="https://agenda-28d7f.firebaseio.com/";
```



- ❖ Luego se instancia un objeto de tipo Firebase

```
private Firebase firebasedatos;
```

- ❖ En el onCreate se configura el contexto de la aplicación y se inicia el objeto Firestore apuntando a la base de datos

```
private String FIREBASE_URL="https://agenda-28d7f.firebaseio.com/";  
private Firestore firebasedatos;
```

- ❖ Quedando:

```
public class MainActivity extends AppCompatActivity implements View.OnClickListener {  
  
    EditText eId, eNombre, eTelefono, eCorreo;  
  
    String nombre, correo, telefono, codigo;  
    Button bInsertar, bActualizar, bBorrar, bBuscar, bLimpiar;  
    Integer id=0;  
  
    private String FIREBASE_URL="https://agenda-28d7f.firebaseio.com/";  
    private Firestore firebasedatos;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        Firestore.setAndroidContext(this);  
        firebasedatos = new Firestore(FIREBASE_URL);  
    }  
}
```

## ❖ Para insertar un dato

```
case R.id.bInsertar:  
    Contacto contacto = new Contacto(nombre, telefono, correo, String.valueOf(id));  
    firebd = firebasedatos.child("contacto "+id);  
    firebd.setValue(contacto);  
    id++;  
    break;
```

## ❖ Para borrar

```
case R.id.bBorrar:  
    firebd = firebasedatos.child("contacto "+codigo);  
    firebd.removeValue();  
    break;
```

## ❖ Para Actualizar

```

case R.id.bActualizar:
    firebd = firebaseDatos.child("contacto "+codigo);

    Map<String, Object> nuevonombre = new HashMap<>();
    nuevonombre.put("nombre", nombre);
    firebd.updateChildren(nuevonombre);

    Map<String, Object> nuevotelefono = new HashMap<>();
    nuevonombre.put("telefono", telefono);
    firebd.updateChildren(nuevotelefono);

    Map<String, Object> nuevocorreo = new HashMap<>();
    nuevonombre.put("mail", correo);
    firebd.updateChildren(nuevocorreo);

    break;
    
```

## ❖ Para Buscar

```
case R.id.bBuscar:
    final String code = "contacto "+codigo;
    firebaseDatos.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            if (dataSnapshot.child(code).exists()) {
                Log.d("data", dataSnapshot.child(code).getValue().toString());
            }
        }

        @Override
        public void onCancelled(FirebaseError firebaseError) {
        }
    });
    break;
```



- ❖ Si se quiere tener acceso a la información de cada uno de los datos se crea un ArrayList de Contactos

```
ArrayList<Contacto> info; //Arreglo para almacenar los datos de Firestore
```

```
private String FIREBASE_URL="https://agenda-28d7f.firebaseio.com/";  
private Firestore firebasedatos;
```

```
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);
```

```
    info = new ArrayList<Contacto>();
```



- ❖ Al hacer la búsqueda se agrega al arrayList y luego se puede acceder a cualquier dato

```
case R.id.bBuscar:
    final String code = "contacto "+codigo;
    firestore.addValueEventListener(new ValueEventListener() {
        @Override
        public void onDataChange(DataSnapshot dataSnapshot) {
            if (dataSnapshot.child(code).exists()) {
                info.add(dataSnapshot.child("contacto "+codigo).getValue(Contacto.class));
                Log.d("data", dataSnapshot.child(code).getValue().toString());
                eNombre.setText(info.get(0).getNombre());
                eTelefono.setText(info.get(0).getTelefono());
                eCorreo.setText(info.get(0).getMail());
            }
        }
        @Override
        public void onCancelled(FirebaseError firebaseError) {
        }
    });
    break;
```