



Firebase

Login con **Android**

Java Day Guatemala 2017
Kevin Villatoro

Contacto: josue.kevin4@gmail.com

goo.gl/fih4WT

Características

Compila apps rápido, sin administrar la infraestructura.

Con el respaldo de Google y la confianza de apps reconocidas.

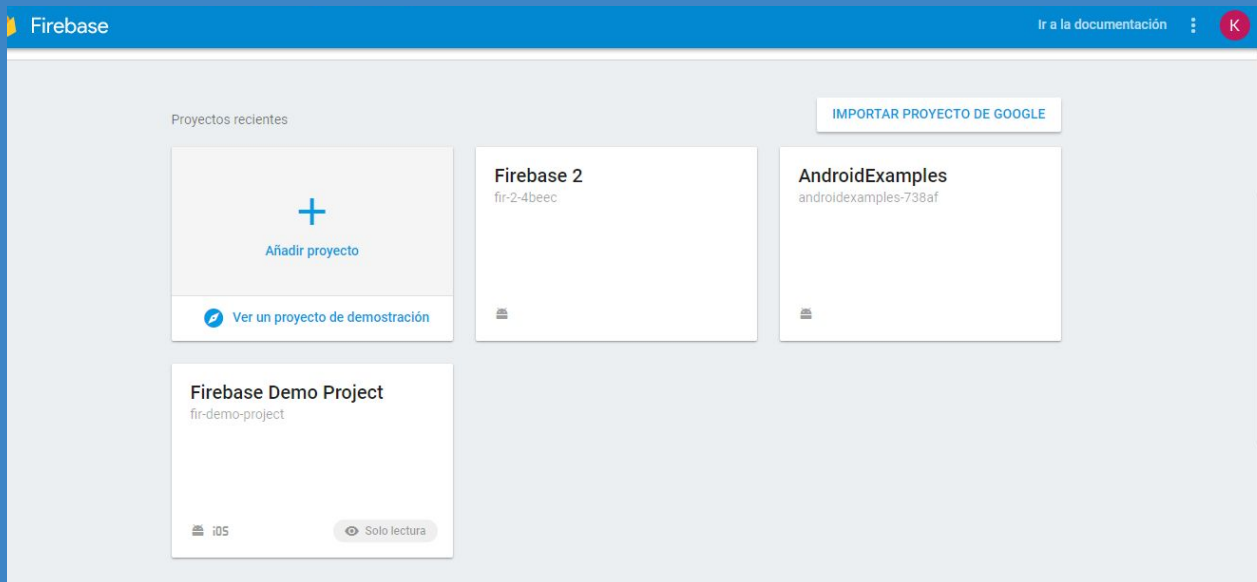
Una consola con productos que funcionan en conjunto.

Primeros pasos:

1) Entrar a la consola de Firebase

<https://console.firebase.google.com>

2) Crear un nuevo proyecto




goo.gl/fih4WT

Crear proyecto

Nombre del proyecto

ID del proyecto ⓘ

nombr-d8586 

País/Región ⓘ

Estados Unidos ▼

De forma predeterminada, tus datos de Analytics se utilizarán para mejorar otras funciones de Firebase, así como otros productos de Google. En la configuración puedes controlar cómo se comparten dichos datos en cualquier momento. [Más información](#)

CANCELAR **CREAR PROYECTO**

The screenshot displays the Firebase console interface. At the top, the Firebase logo and 'JavaDay' are on the left, and a link to 'Ir a la documentación' (Go to documentation) and a user profile icon 'K' are on the right. The left sidebar contains navigation links: Overview (selected), Analytics, and a 'DEVELOP' section with Authentication, Database, Storage, Hosting, Functions, Test Lab, Crashlytics, Crash Reporting, and Performance. The main content area is titled 'Visión general' and features a welcome message: 'Te damos la bienvenida a Firebase. Empieza aquí.' Below this, there are three colored circles with icons: a blue circle with 'iOS', a green circle with the Android robot, and a red circle with code symbols '</>'. Each circle has a corresponding text prompt: 'Añade Firebase a tu aplicación de iOS', 'Añade Firebase a tu aplicación de Android', and 'Añade Firebase a tu aplicación web'. A notification banner at the top of the main area offers to receive emails related to Firebase features, research, and events, with a 'REGISTRARSE' (Sign up) link and a close button 'X'.

```
keytool -list -v -keystore  
"C:\user_name\.android\debug.keystore"  
-alias androiddebugkey -storepass  
android -keypass android
```

```
keytool -list -v -keystore
.android/debug.keystore -alias
androiddebugkey -storepass android
-keypass android
```

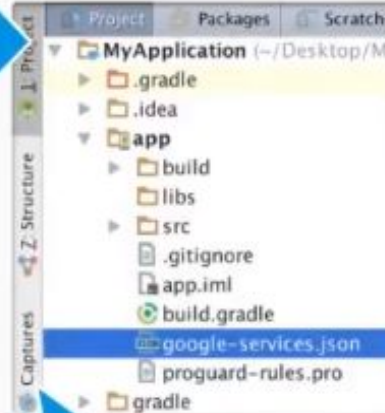
Instrucciones para Android Studio

Alternativas: [Unity](#) [C++](#)

1.  **Descargar google-services.json**

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

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

**google-services.json**

Añade Firebase a tu aplicación de Android

1 Registrar la aplicación 2 Descargar el archivo de configuración 3 Añadir el SDK de Firebase

El complemento de los servicios de Google para [Gradle](#) carga el archivo `google-services.json` que acabas de descargar. Para poder usar el complemento, debes modificar los archivos `build.gradle`.

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

```
buildscript {
    dependencies {
        // Add this line
        classpath 'com.google.gms:google-services:3.1.0'
    }
}
```
2. `build.gradle` de aplicación (`<project>/<app-module>/build.gradle`):

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

incluye Analytics de forma predeterminada ⓘ
3. Por último, pulsa **Sincronizar ahora** en la barra que aparece en el entorno IDE (entorno de desarrollo integrado):

Gradle files have changed since... **Sync now**

Configuración Finalizada

Android Studio

Clases a usar:

FirebaseAuth

FirebaseAuth.AuthStateListener

Android Studio

```
private void inicializar() {  
    firebaseAuth = FirebaseAuth.getInstance();  
    authStateListener = (AuthStateListener) (firebaseAuth) → {  
        FirebaseUser firebaseUser = firebaseAuth.getCurrentUser();  
        if(firebaseUser !=null){  
            Log.w(TAG, "onAuthStateChanged-signed-in" + firebaseUser.getId());  
            Log.w(TAG, "onAuthStateChanged-signed-in" + firebaseUser.getEmail());  
        }else{  
            Log.w(TAG, "onAuthStateChanged-log-out" );  
        }  
    };  
}
```

Android Studio

```
private void crearCuenta(final String correo, String pass){  
    firebaseAuth.createUserWithEmailAndPassword(correo,pass).addOnCompleteListener((task) -> {  
        if(task.isSuccessful()){  
            Toast.makeText(MainActivity.this,"Creado exitosamente, puede iniciar sesion",Toast.LENGTH_SHORT).show();  
        }else if(firebaseAuth.getCurrentUser().getEmail().equals(correo)){  
            Toast.makeText(MainActivity.this,"Ya existe este correo registrado",Toast.LENGTH_SHORT).show();  
        }else {  
            Toast.makeText(MainActivity.this,"No se pudo crear la cuenta",Toast.LENGTH_SHORT).show();  
        }  
    });  
}
```

Android Studio

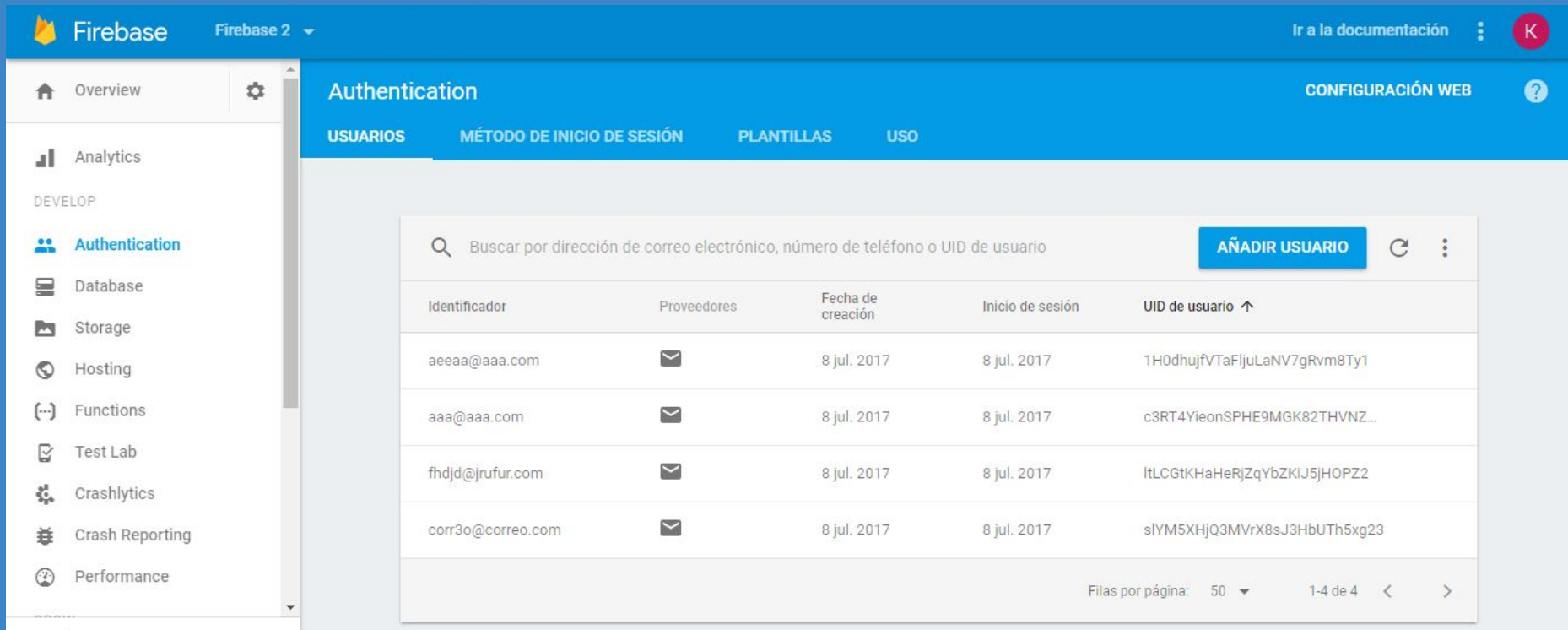
```
private void iniciarSesion(String correo,String pass){  
    firebaseAuth.signInWithEmailAndPassword(correo,pass).addOnCompleteListener({task} → {  
        if(task.isSuccessful()){  
            Toast.makeText(MainActivity.this,"Inicio correcto",Toast.LENGTH_SHORT).show();  
        }else{  
            Toast.makeText(MainActivity.this,"Verifique sus datos",Toast.LENGTH_SHORT).show();  
        }  
    });  
}
```

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    inicializar();
    btnCrearCuenta = (Button)findViewById(R.id.btn_Create);
    btnInicar = (Button)findViewById(R.id.btn_SigIn);
    txtCorreo = (EditText)findViewById(R.id.txt_user);
    txtPassword = (EditText)findViewById(R.id.txt_pass);

    btnCrearCuenta.setOnClickListener((view) -> {
        if(txtCorreo.getText().toString().isEmpty() || txtPassword.getText().toString().isEmpty()){
            Toast.makeText(MainActivity.this, "Debe llenar los campos", Toast.LENGTH_SHORT).show();
        } else if(txtPassword.getText().length() <= 6){
            Toast.makeText(MainActivity.this, "La contraseña debe tener mas de 6 caracteres", Toast.LENGTH_SHORT).show();
        }
        else{
            crearCuenta(txtCorreo.getText().toString(), txtPassword.getText().toString());
        }
    });

    btnInicar.setOnClickListener((view) -> {
        if(txtCorreo.getText().toString().isEmpty() || txtPassword.getText().toString().isEmpty()){
            Toast.makeText(MainActivity.this, "Debe llenar los campos", Toast.LENGTH_SHORT).show();
        } else{
            iniciarSesion(txtCorreo.getText().toString(), txtPassword.getText().toString());
        }
    });
}
```

Android Studio



The screenshot shows the Firebase Authentication console. The left sidebar contains navigation links for Overview, Analytics, Authentication (selected), Database, Storage, Hosting, Functions, Test Lab, Crashlytics, Crash Reporting, and Performance. The main content area is titled 'Authentication' and includes tabs for USUARIOS, MÉTODO DE INICIO DE SESIÓN, PLANTILLAS, and USO. A search bar at the top of the table allows filtering by email, phone number, or UID. A table lists five users with columns for Identifier, Providers, Creation Date, Session Start, and UID. At the bottom right, there are controls for rows per page (50) and page navigation (1-4 de 4).

Identificador	Proveedores	Fecha de creación	Inicio de sesión	UID de usuario ↑
aeeaa@aaa.com	✉	8 jul. 2017	8 jul. 2017	1H0dhujfVTaFljuLaNV7gRvm8Ty1
aaa@aaa.com	✉	8 jul. 2017	8 jul. 2017	c3RT4YieonSPHE9MGK82THVNZ...
fhdjd@jrufur.com	✉	8 jul. 2017	8 jul. 2017	ltLCGtKHaHeRjZqYbZKIj5jHOPZ2
corr3o@correo.com	✉	8 jul. 2017	8 jul. 2017	slYM5XHjQ3MvRX8sJ3HbUTh5xg23

goo.gl/fih4WT

Video: goo.gl/Kh3v9k