



Firebase

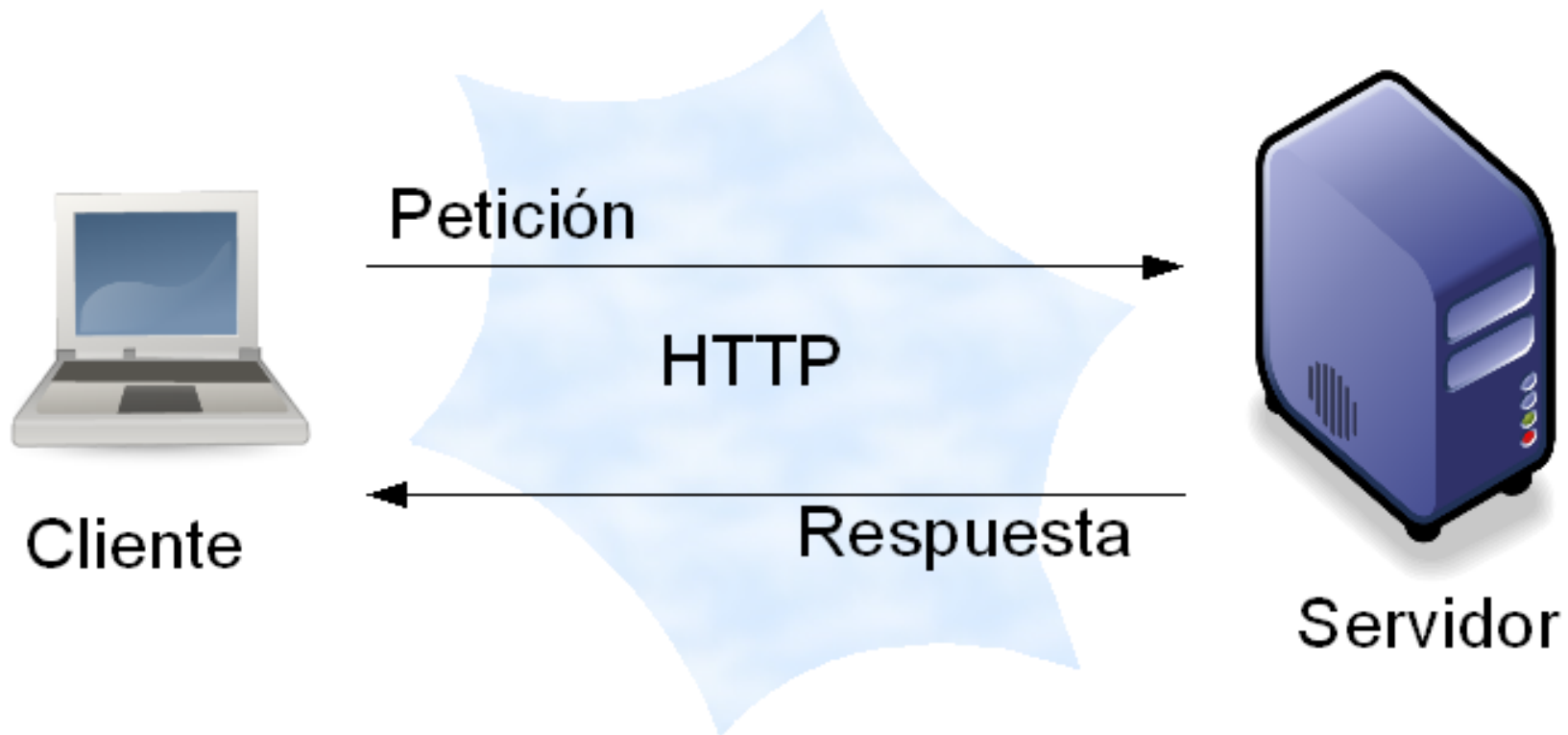
Contenidos

1. Conocimientos previos.
2. ¿Qué es la nube de Google(Google cloud)?
3. ¿Qué es Firebase?¿En que se basa?
4. Ventajas/Desventajas SQL-NoSQL.
5. ¿Utilizamos una BBDD SQL o NOSQL?
6. Definición genérica de tiempo real.
7. Introducción a Firebase RT-Database.
8. Características de Firebase RT-Database.
9. Implementación de Firebase RT-Database.
10. Casos reales de Apps con Firebase.
11. ¿Preguntas?

1. Conocimientos previos.

- NoSQL
- SQL
- Protocolo OAuth
- Programación concurrente/paralela.
- UI/UX
- Estadística básica.
- Gradle, Cocoapods, etc.
- JSON
- Arquitectura Cliente/Servidor

1. Conocimientos previos.



2.La nube de Google



Google Cloud Platform

2.La nube de Google



Application
Runtime Services

Enabling No-Touch Operations



Data Services

Breakthrough Insights,
Breakthrough Applications



Foundation
Infrastructure & Operations

The Gear that Powers Google



3.¿Qué es Firebase?

- **Firestore** es una plataforma de desarrollo multiplataforma en la nube de Google.


3 ¿Qué es firebase?



Project Overview




DEVELOP

 Authentication

Database

Storage

 Hosting

(...) Functions

STABILITY

 Crashlytics

Crash Reporting

Performance

 **Test Lab**

Spark
Gratis USD 0 por mes

ACTUALIZAR

SpotApp ▼

[Ir a la documentación](#)



Descripción general



1 app en el proyecto

AGREGAR OTRA APP



akeen.app.SpotApp

Usuarios activos por día

0.



Usuarios activos por mes

11.



Usuarios que no experimentaron bloqueos

100 %



Bloqueos

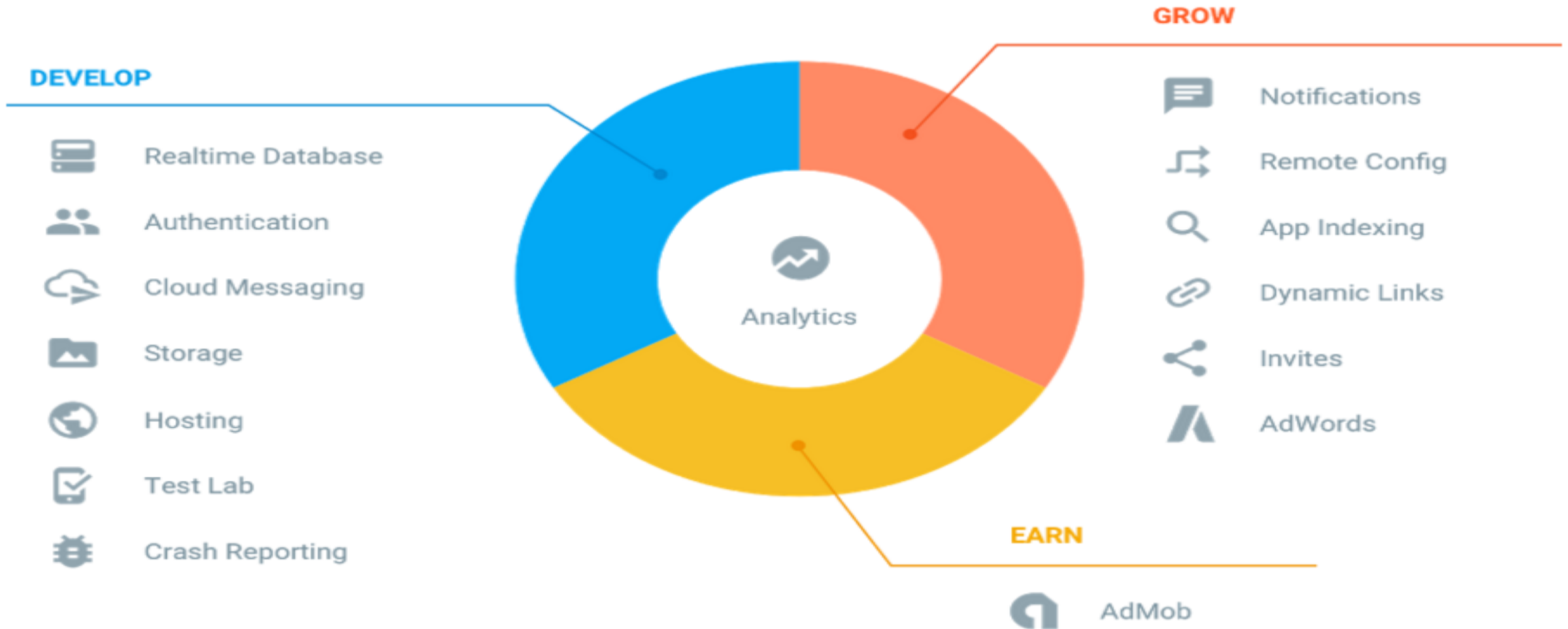
0



3.¿En que se basa?

1. Desarrollo
2. Crecimiento
3. Obtención de beneficios
4. Estabilidad(nueva).

3.¿En que se basa?



4.NoSQL

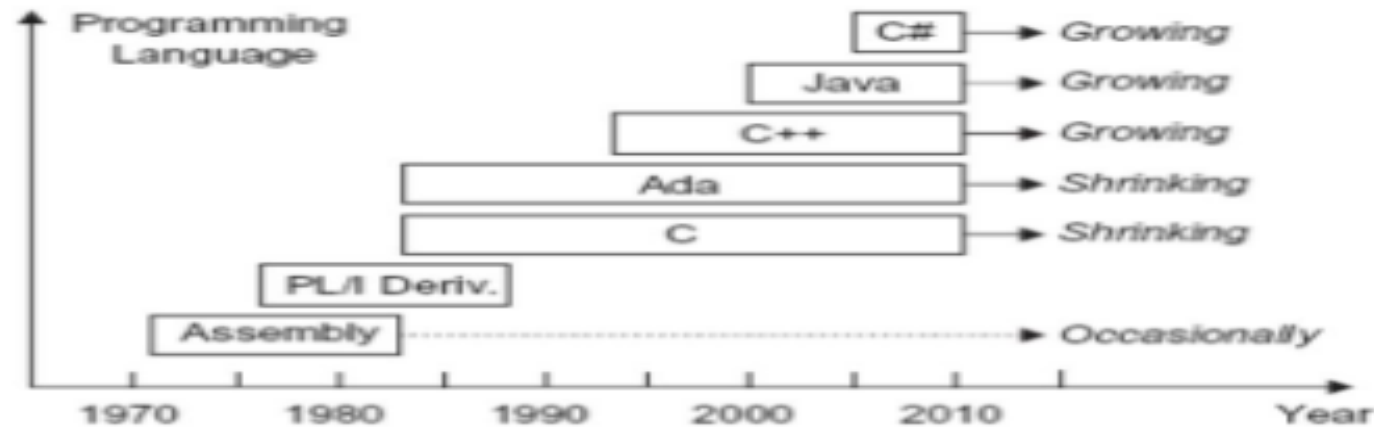
Ventajas	Desventajas
<p>adaptan mejor que el modelo relacional.</p> <p>máquinas con pocos recursos.</p> <p>Optimización de consultas</p> <p>NoSQLQies a través de eventos asíncronos.</p>	<p>Problemas de compatibilidad SQL</p> <p>Falta de estandarización.</p> <p>herramientas de administración no muy usables o consola.</p> <p>Hay que tener conocimiento previos de programación concurrente y paralela.</p> <p>App concurrentes difíciles de desarrollar.</p>

5.¿Cuando utilizar SQL o NoSQL?

SQL	NoSQL
<p>Cuando los datos deben ser consistentes</p> <p>Elegir sin ninguna duda si necesitamos modelo relacional.</p>	<p>Cuando nuestro presupuesto no se puede permitir grandes máquinas y debe destinarse a máquinas de menor rendimiento.</p> <p>Cuando las estructuras de datos que manejamos son variables.</p> <p>Análisis de grandes cantidades de datos en modo lectura.</p> <p>Captura y procesamiento de eventos.</p>

6.¿A tiempo real? R.T.-Database/R.T.-Programming

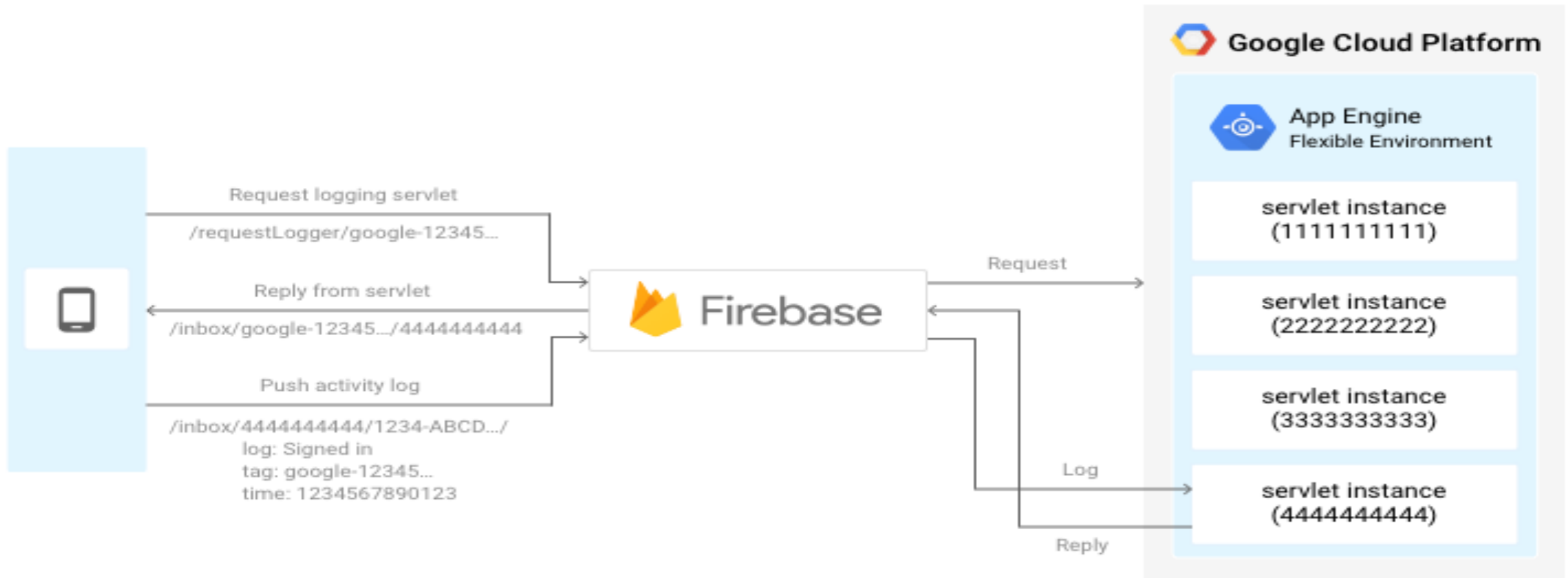
- Fijamos el tiempo en que dicha consulta se ejecuta.
- Unidad temporal: ms, s, m, h, d, y.
- Este concepto es el mismo para la ejecución de algoritmos concurrentes R.T.



7. Realtime RT-Database

- **Realtime Database** es la base de datos original de Firebase. Es una solución eficiente de **baja latencia** para las apps para dispositivos móviles que necesitan estados sincronizados entre los clientes en tiempo real.
- **NoSQL Database** es una amplia clase de sistemas de gestión de bases de datos que difieren del modelo clásico de SGBDR .Los datos almacenados no requieren estructuras fijas como tablas, normalmente no soportan operaciones JOIN, ni garantizan completamente ACID(atomicity,consistency,isolation,durability).

7. Firebase RT-Database



7. Firebase RT-Database

https://final-project-recording.firebaseio.com/

final-project-recording

courses

lessons

-KT_udWS6pEmpLVrxlVw

courseId: "-KT_udWMM0vKlnp-naE3"

description: "Angular 2 Tutorial For Beginners - Build Your F..."

duration: "2:49"

longDescription: "This is step by step guide to create your first..."

tags: "BEGINNER"

url: "angular2-hello-world-write-first-application"

videoUrl: "https://www.youtube.com/embed/du6sKwEFrhQ"

-KT_udWT2-3eSndrJMiK

-KT_udWT2-3eSndrJMiL

-KT_udWUCB2RDtUT59y2

```
{
  "users": {
    "user1": {
      "price": "200"
    },
    "user2": {
      "price": "50"
    },
    "user3": {
      "price": "100"
    }
  },
  "advisors-countries": {
    "user1": {
      "destinations": [
        {
          "name": "Angola"
        },
        {
          "name": "Australia"
        }
      ]
    },
    "user2": {
      "destinations": [
        {
          "name": "Angola"
        },
        {
          "name": "Austria"
        },
        {
          "name": "Jordan"
        }
      ]
    },
    "user3": {
      "destinations": [
        {
          "name": "Egypt"
        },
        {
          "name": "Austria"
        },
        {
          "name": "Jordan"
        }
      ]
    }
  }
}
```


8.Características Firebase RT-Database

- Tiempo real.
- Sin conexión.
- Accesible desde dispositivos Cliente.
- Escalamiento en varias BBDD.(*plan de precios Blaze).

9.Implementación de Firebase RT-Database

- Integrar los SDK de Firebase RT-Database.
- Crear referencias de RT-Database.
- Configurar datos y escuchar para detectar cambios.
- Habilitar la persistencia sin conexión.
- Proteger los datos.(OAuth)

9.1 Implementación de Firebase RT-Database

[Productos](#)[Casos de uso](#)[Precios](#)[Documentos](#)[Asistencia](#)[IR A LA CONSOLA](#)

Documentación

[DESCRIPCIÓN GENERAL](#)[GUÍAS](#)[REFERENCIA](#)[MUESTRAS](#)[BIBLIOTECAS](#)[ENVIAR COMENTARIOS](#)

Anuncio de Cloud Firestore (Beta): Prueba la nueva base de datos escalable y flexible de Firebase y Google Cloud Platform. [Obtén más información sobre Cloud Firestore.](#)

Realtime Database

Introducción

Selecciona una base de datos

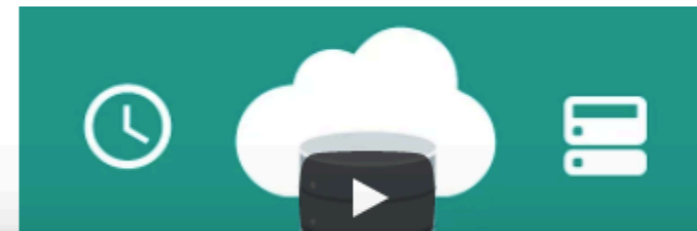
- › iOS
- › Android
- › Web
- › Admin
- › REST
- › C++
- › Unity
- › Seguridad y reglas
- › Uso y rendimiento
- Copias de seguridad automáticas
- Extender con Cloud Functions

Firestore Realtime Database



Almacena y sincroniza datos con nuestra base de datos NoSQL alojada en la nube. Los datos se sincronizan con todos los clientes en tiempo real y se mantienen disponibles cuando tu app está sin conexión.

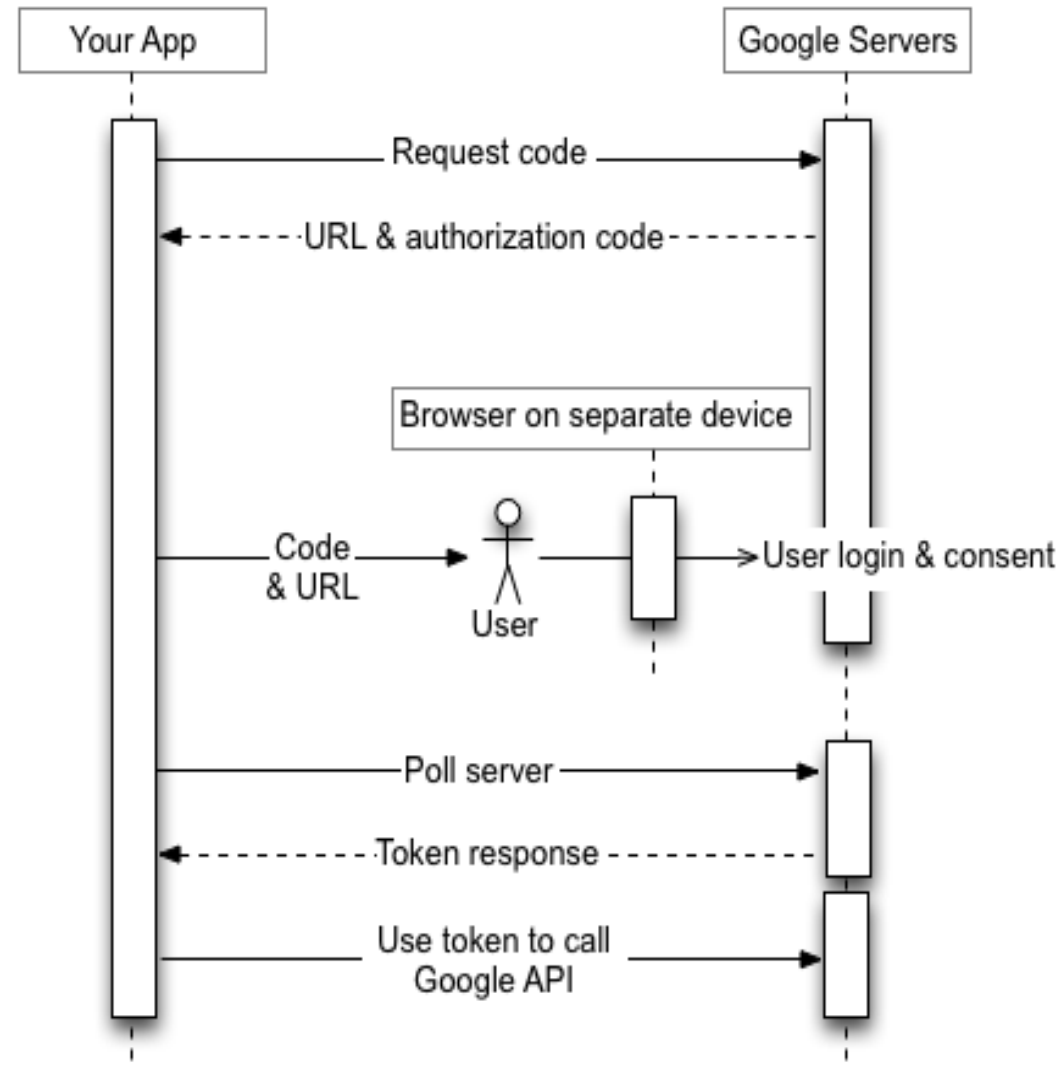
Firestore Realtime Database es una base de datos alojada en la nube. Los datos se almacenan en formato JSON y se sincronizan en tiempo real con cada cliente conectado. Cuando compilas apps multiplataforma con nuestros



Contenido

[Funciones clave](#)[¿Cómo funciona?](#)[Ruta de implementación](#)[¿Quieres almacenar otros tipos de datos?](#)[Próximos pasos:](#)

9.2 Proteger datos (OAuth)



9.2 Proteger datos



The screenshot shows the Firebase console interface. On the left sidebar, the 'Security & Rules' option is highlighted with a red arrow. The main content area is titled 'FIREBASE RULES' and displays a JSON configuration for database security rules. The rules are as follows:

```
1 {  
2   "rules": {  
3     ".read": false,  
4     ".write": false,  
5  
6     "users": {  
7       "$userId": {  
8         "profile": {  
9           ".read": "(auth != null) && ($userId === auth.uid)",  
10          ".write": "(auth != null) && ($userId === auth.uid)"  
11        }  
12      }  
13    }  
14  }  
15  
16 }
```

10.Casos reales de Apps con firebase

- Shazam
- Fabulous
- Sky-scanner
- Tune-in
- [Alibaba](#)
- pic-collage.com

¿Preguntas?



<https://Joaquinpco.github.io>