

Indice ;

Tabla de contenido

[Introduccion : 4](#_Toc165231894)

[2. ¿Cuál es la definición de Cloud Firestore? 4](#_Toc165231895)

[Características clave: 5](#_Toc165231896)

[Consultas y Flexibilidad de Búsqueda: 5](#_Toc165231897)

[Las principales diferencias entre las bases de datos SQL y NoSQL: 6](#_Toc165231898)

[Consistencia y Transacciones: 6](#_Toc165231899)

[Conclusión: 6](#_Toc165231900)

[Capturas: 7](#_Toc165231901)

[8](#_Toc165231902)

# Introduccion :

Muchas aplicaciones modernas requieren un manejo eficiente de datos en tiempo real, y Firebase de Google ofrece dos soluciones principales para este problema: Cloud Firestore y Database Realtime. Ambos servicios tienen características, ventajas y usos distintos. Este trabajo examinará en detalle Cloud Firestore y Realtime Database, discutiendo sus diferencias fundamentales, comparando bases de datos SQL y NoSQL y decidiendo cuál sería la mejor opción para implementar en una aplicación desarrollada en React Native.

2. ¿Cuál es la definición de Cloud Firestore?   
 Definición y funcionamiento: Cloud Firestore es una base de datos NoSQL en tiempo real que almacena datos en documentos JSON dentro de colecciones. Permite consultas avanzadas y utiliza una estructura de datos flexible.   
  
  
 Características destacadas:   
  
Flexible estructura de datos   
Preguntas Avanzadas   
Autoescalabilidad   
Sync en tiempo real   
Sistema de seguridad integrado   
¿Qué es una base de datos en tiempo real?   
Definición y aplicación:   
Los datos se sincronizan instantáneamente entre los clientes y los servidores a través de una base de datos en tiempo real. sincroniza y almacena los datos en una estructura JSON en tiempo real.

Características clave:   
  
Formato de datos JSON en tiempo real sincronizado   
  
Escalabilidad reducida   
Las reglas de seguridad que se pueden personalizar   
 Las diferencias entre Cloud Firestore y la base de datos en tiempo real:   
Estructura de datos y modelado: Cloud Firestore ofrece una estructura de datos más adaptable que admite tipos de datos complejos y objetos anidados. La base de datos realtime utiliza una estructura JSON plana.   
  
 Escalabilidad y rendimiento: Cloud Firestore proporciona escalabilidad automática y mejora la gestión de cargas de trabajo cambiantes. La escalabilidad de Realtime Database es limitada, especialmente para aplicaciones que requieren mucha escritura.

Consultas y Flexibilidad de Búsqueda:

Cloud Firestore ofrece consultas de búsqueda avanzadas y flexibilidad. Las capacidades de consulta de la base de datos en tiempo real son más limitadas.  
  
  
 Sincronización en tiempo real y Seguridad: Aunque ambos servicios ofrecen sincronización en tiempo real, Cloud Firestore ofrece una seguridad más avanzada a través de reglas de seguridad más avanzadas.

Las principales diferencias entre las bases de datos SQL y NoSQL:   
Modelado de Datos y Flexibilidad:

El modelo de datos de las bases de datos SQL es estructurado y rígido, mientras que las bases de datos NoSQL ofrecen una mayor flexibilidad.   
  
 Escalabilidad y Rendimiento: Las bases de datos NoSQL generalmente son más escalables y funcionan mejor en entornos distribuidos que las bases de datos SQL.

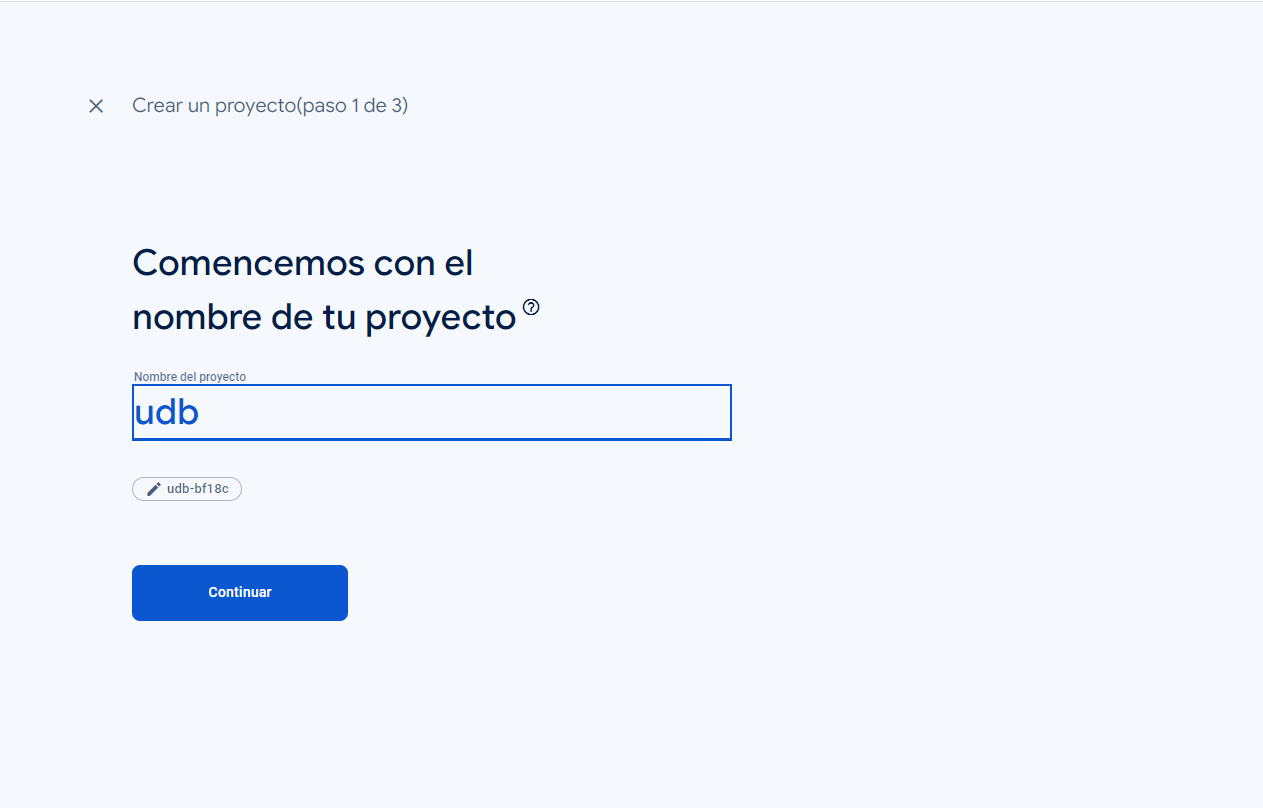
Consistencia y Transacciones: Las bases de datos SQL tienen una mayor consistencia y soportan transacciones ACID, mientras que las bases de datos NoSQL pueden tener modelos de consistencia más antiguos y no soportan transacciones ACID.   
  
  
¿Cuál es el método más efectivo para aplicar en una aplicación hecha con React Native?  
Exigencias y casos de uso:   
  
La elección entre Realtime Database y Cloud Firestore dependerá de las necesidades y casos de uso de cada aplicación.   
  
Consideraciones relacionadas con el desarrollo y el mantenimiento: La estructura de datos flexible de Cloud Firestore y la capacidad de consulta avanzada facilitan la expansión y el mantenimiento.

# Conclusión:

Dependiendo de la necidad se puede usar uno u otro lo cierto es que el servicio de Google tanto de firebase realtime database como el fire estore ofrece una salida para los programadores facilitando mucho el Back end

A diferencia de una base de datos relacional como SQL que requiere mas trabajo pero a la vez es mas solida y a parte de soportar mayor volumen de información también se pueden hacer varias cosas como “gatillos” “procedimientos almacenado” “vistas” entre otros

# Capturas:

sss

# 

