



Universidad Don Bosco

Foro Discucion

presenta:

Marco José Mejía Tario	MT172007
Diego Antonio Fuentes Guardado	FG230381
Rene Emanuel Lopez Maldonado	LM211126

Fecha
28/04/2024

ÍNDICE

Introducción

2 Cloud Firestore

2.1 ¿Qué es Cloud Firestore?

2.2 Características principales de Cloud Firestore

2.3 Beneficios de usar Cloud Firestore

3 Realtime Database

3.1 ¿Qué es Realtime Database?

3.2 Características principales de Realtime Database

3.3 Beneficios de usar Realtime Database

3.4 Diferencias entre Cloud Firestore y Realtime Database

3.5 Diferencias fundamentales entre bases de datos SQL y NoSQL

3.6 ¿Cuál base de datos es mejor para una aplicación React Native?

7 base de datos nosql con Firestore.

4 Conclusión

INTRODUCCIÓN

La elección de la base de datos adecuada en el mundo del desarrollo de aplicaciones móviles es un factor crítico que determina el rendimiento, la escalabilidad y la experiencia general del usuario. Cloud Firestore y Realtime Database son dos bases de datos NoSQL populares que están disponibles para Firebase.

El objetivo de este artículo es analizar y comparar estas dos bases de datos, examinando sus características, diferencias y beneficios para las aplicaciones React Native. Se espera que la información proporcionada ayude a tomar una decisión informada sobre la base de datos más apropiada para un proyecto en particular.

2 Cloud Firestore

2.1 ¿Qué es Cloud Firestore?

Cloud Firestore es una base de datos NoSQL escalable y adaptable que se ofrece a través de Cloud Firebase.



Su objetivo es almacenar y sincronizar datos en tiempo real para aplicaciones web y móviles. Cloud Firestore utiliza un modelo de datos basado en documentos, lo que facilita las consultas y organiza la información de manera jerárquica.

2.2 Características principales de Cloud Firestore

- **Modelo de datos flexible:** Los datos se pueden almacenar en Cloud Firestore como documentos JSON con campos anidados y subcolecciones. Esto facilita la organización y recuperación de datos complejos.
- **Consultas potentes:** El lenguaje de consultas flexible de Cloud Firestore permite filtrar, ordenar y limitar los resultados según una variedad de criterios.
- **Escalabilidad sin servidor:** Cloud Firestore se expande automáticamente para adaptarse a cualquier carga de trabajo, sin necesidad de administrar la infraestructura de la base de datos.
- **Sincronización en tiempo real:** La experiencia de usuario es fluida y en tiempo real porque todos los clientes conectados ven automáticamente los cambios en los datos.
- **Soporte sin conexión:** Incluso cuando el dispositivo no está conectado a Internet, Cloud Firestore permite almacenar y acceder a datos.
- **Seguridad robusta:** Los controles de seguridad granulares de Cloud Firestore protegen los datos de la aplicación.

2.3 Beneficios de usar Cloud Firestore

- **Desarrollo rápido:** Con su modelo de datos flexible, consultas potentes y soporte sin conexión, Cloud Firestore facilita el desarrollo de aplicaciones móviles y web.
- **Escalabilidad sin esfuerzo:** La escalabilidad automática de Cloud Firestore permite al desarrollador concentrarse en la lógica de la aplicación.
- **Experiencia de usuario en tiempo real:** La experiencia de usuario es fluida y receptiva porque todos los clientes ven inmediatamente los cambios en los datos.
- **Seguridad confiable:** Los controles de seguridad granulares y el cifrado integrado de Cloud Firestore protegen los datos de la aplicación.

3 Realtime Database

3.1 ¿Qué es Realtime Database?

La base de datos Realtime es la base de datos NoSQL original de Firebase. Es una base de datos que funciona como un árbol JSON que permite almacenar y sincronizar datos en tiempo real. Realtime Database es popular para proyectos pequeños y medianos porque es simple y fácil de usar.

3.2 Características principales de Realtime Database

- **Estructura de datos simple:** La base de datos en tiempo real almacena los datos en un árbol JSON, lo que facilita la comprensión y el acceso a la información.
- **Sincronización en tiempo real:** Garantiza una experiencia de usuario fluida y receptiva porque todos los clientes conectados ven inmediatamente los cambios en los datos.
- **Fácil de usar:** Realtime Database tiene una API fácil de usar y fácil de aprender, lo que la hace atractiva para los desarrolladores principiantes.
- **Escalabilidad:** La base de datos de tiempo real puede escalar horizontalmente para manejar grandes cargas de trabajo.

3.3 Beneficios de usar Realtime Database

- **Desarrollo rápido:** La facilidad de uso de Realtime Database permite el desarrollo rápido de aplicaciones, especialmente para proyectos de tamaño mediano y pequeño.
- **Sincronización en tiempo real robusta:** Para las aplicaciones que requieren actualizaciones de datos en todos los dispositivos, la sincronización en tiempo real de la base de datos es ideal.
- **Facilidad de uso:** Los desarrolladores que no tienen mucha experiencia con bases de datos NoSQL pueden usar la API sencilla de Realtime Database.
- **Comunidad grande:** La base de datos en tiempo real tiene una amplia y activa comunidad de desarrolladores, lo que facilita la búsqueda de ayuda y recursos.

4 Diferencias entre Cloud Firestore y Realtime Database

Característica	Cloud Firestore	Realtime Database
Modelo de datos	Documentos JSON	Árbol JSON
Consultas	Lenguaje de consultas flexible	Consultas básicas
Escalabilidad	Sin servidor, escalabilidad automática	Escalabilidad horizontal
Seguridad	Controles granulares y cifrado	Reglas de seguridad básicas
Costos	Basado en el uso de almacenamiento y operaciones	Basado en el ancho de banda y las conexiones

5 Diferencias fundamentales entre bases de datos SQL y NoSQL

Característica	SQL	NoSQL
Estructura de datos	Tablas relacionales	Estructuras flexibles (documentos, gráficos, etc.)
Consistencia	Fuerte consistencia ACID	Eventual consistency
Esquema	Esquema definido previamente	Esquema flexible
Escalabilidad	Escalabilidad vertical	Escalabilidad horizontal

6 ¿Cuál base de datos es mejor para una aplicación React Native?

Cloud Firestore:

Modelo de datos adaptable: Cloud Firestore utiliza un modelo de datos basado en documentos JSON, lo que permite organizar la información de manera jerárquica y simplificar las consultas. Esto es perfecto para las aplicaciones porque se necesita almacenar datos complejos que están conectados entre sí.

Potentes consultas: el lenguaje de consultas de Cloud Firestore es flexible, lo que permite filtrar, ordenar y limitar los resultados de acuerdo con varios criterios. Esto permite recuperar de manera eficiente y precisa la información que se requiere.

La escalabilidad sin servidor: significa que Cloud Firestore se escala automáticamente para adaptarse a cualquier carga de trabajo, sin necesidad de que se administre la infraestructura de la base de datos. Esto facilita el concentrarse sin preocuparse por problemas de rendimiento a medida que crece la base de usuarios.

Sincronización en tiempo real: para garantizar una experiencia de usuario fluida y en tiempo real, los cambios en los datos se reflejan automáticamente en todos los clientes conectados. Esto es fundamental para las aplicaciones porque se necesita que los usuarios siempre tengan acceso a la información más reciente.

Soporte sin conexión: Cloud Firestore permite el almacenamiento y acceso a datos incluso cuando el dispositivo no está conectado a Internet. Esto es crucial para la aplicación porque se necesita que las personas puedan usarla incluso si no tienen una conexión a Internet estable.

Seguridad sólida: Cloud Firestore proporciona controles de seguridad granulares para proteger los datos de la aplicación. Esto brinda la confianza para que los usuarios no tengan nada que perder.

En general, Cloud Firestore nos brinda la flexibilidad y las características necesarias para desarrollar una aplicación React Native escalable, segura y con una experiencia de usuario excelente.

7. base de datos con nosql con Firestore.

Alcance de la consulta		Ruta de acceso ?
Colección		/Alumnos
+ Agregar a la consulta		
Resultados de la consulta		
Document ID	Carrera	Nombre
2PEH9yxRVwY7osHgOk5H	"Técnico en Ingeniería en Computación"	"Diego Antonio Fuentes Guardado"
3oPyc9q2NDWztw4ccJhK	"Técnico en Ingeniería en Computación"	"rene emanuel lopez maldonado"
IU2o0NGu1qchOTvPyOS7	"Técnico en Ingeniería en Computación"	"Marco José Mejía Tario"
Kq56RjtnLJa4cOPZpOMw	"Técnico en Ingeniería en Computación"	"Alisson Rebeca Ramos Sibrian"
eapK6e54YNPMdYjfGj3W	"Técnico en Ingeniería en Computación"	"Hugo Manrique Hernández García"

Vista de la tabla alumnos

Alcance de la consulta		Ruta de acceso ?
Colección		/Profesores
+ Agregar a la consulta		
Resultados de la consulta		
Document ID	Asignatura	Nombre
AgKEGi3SXZw7Wy9Vf7mT	"Historia de la Edad Media"	"Rosa Gómez"
IFrNDru53unAYtmZHqZ	"Diseño y Programación de Software Multiplataforma"	"Alexander Alberto Siguenza Campos"
OPd0dS2vrg1sTVikhRW1	"Literatura hispanoamericana"	"Pedro Sánchez"
IVogaXQscZaXMLWjeW1h	"Cálculo I"	"Carlos Rodríguez"
zeDPn7CkjS8jHFh1RAh	"Química orgánica"	"Ana García"

Vista de la tabla profesores

Alcance de la consulta

Ruta de acceso

Colección

/Asignaturas

Agregar a la consulta

Resultados de la consulta

Document ID	Nombre
FJpgKqMoAmT7EJCu9wVa	"Química orgánica"
OOeVF2o4nvVR3qFWLKTW	"Historia de la Edad Media"
hifysxH2ddZZTm0i6qqp	"Diseño y Programación de Software Multiplataforma"
j9VySGE7L6fwcSyzUFkZ	"Literatura hispanoamericana"
oHkZ7mVMh1sKyhG5tHc5	"Cálculo I"

Vista de la tabla asignaturas

Alcance de la consulta

Ruta de acceso

Colección

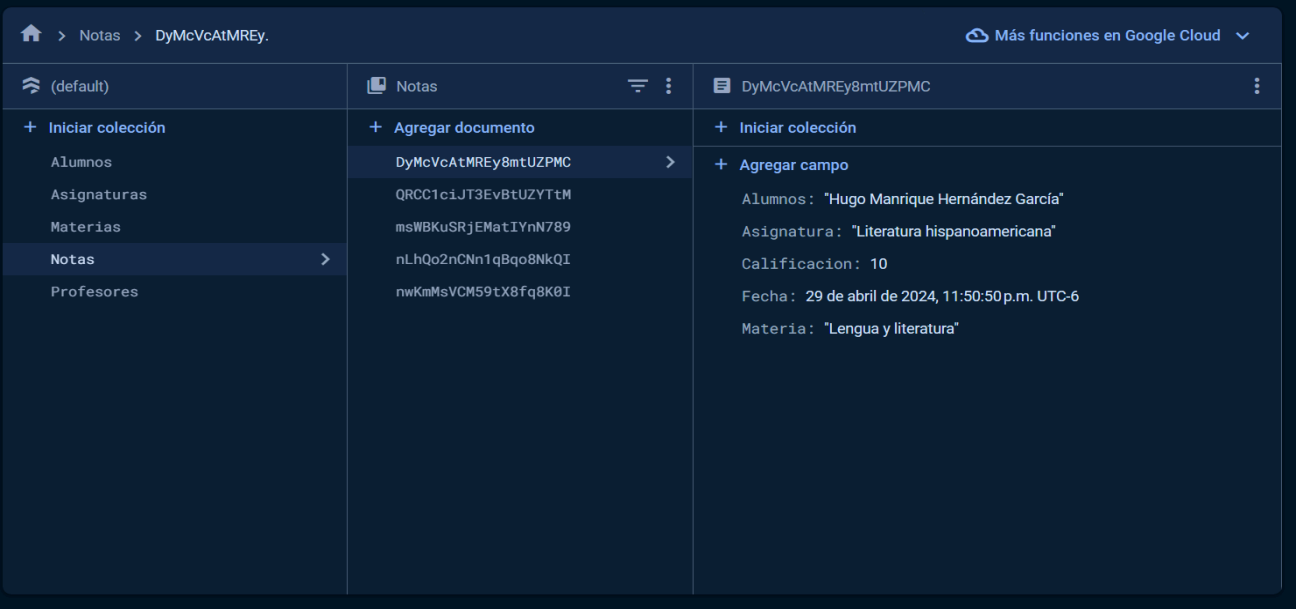
/Notas

Agregar a la consulta

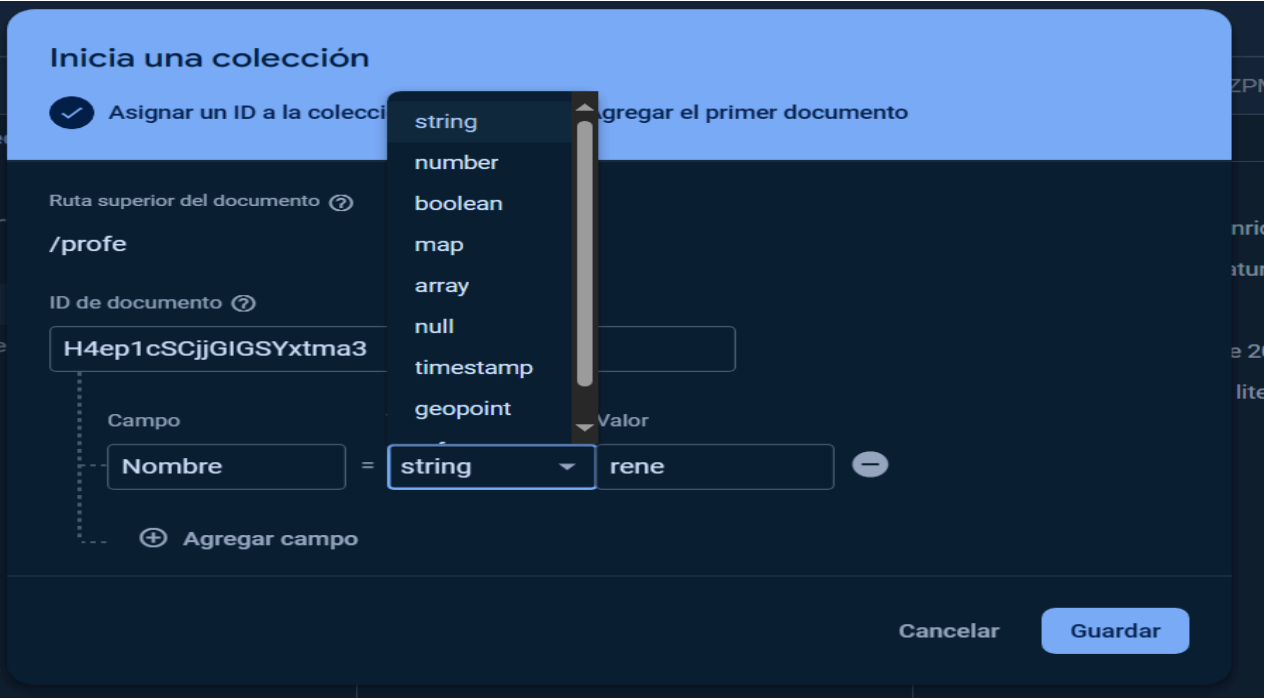
Resultados de la consulta

Document ID	Alumnos	Asignatura	Calificacion	Fecha	Materia
DyMcVcAtmREy8mtUZPMC	"Hugo Manrique Hernández García"	"Literatura hispanoamericana"	10	29 de abril de 2024, 11:50:50 p.m. UTC-6	"Lengua y literatura"
QRCC1ciJT3EvBtUZYTtM	"Marco José Mejía Tario"	"Cálculo I"	9	28 de abril de 2024, 11:50:50 p.m. UTC-6	"Matemáticas"
msWBKuSRjEMatIYnN789	"rene emanuel lopez maldonado"	"Diseño y Programación de Software Multiplataforma"	10	28 de abril de 2024, 11:50:50 p.m. UTC-6	"Progamacion"
nLhQo2nCNn1qBqo8NkQI	"Alisson Rebeca Ramos Sibrian"	"Historia de la Edad Media"	8	30 de abril de 2024, 11:50:50 p.m. UTC-6	"Historia"
nwKrmMsVCM59tX8fq8K0I	"Diego Antonio Fuentes Guardado"	"Química orgánica"	9	29 de abril de 2024, 11:50:50 p.m. UTC-6	"Ciencias naturales"

Vista de la tabla notas



Inserción de datos



Inicio de coleccion en tabla profesores

CONCLUSIÓN

Las tecnologías de bases de datos Nosql ofrecidas por Firebase se analizaron de manera minuciosa, tanto sus características y beneficios además de sus diferencias comparadas con las bases de datos SQL tradicionales, por otra parte, las bases de datos SQL tradicionales no son la mejor opción para la gran mayoría de aplicaciones de React Native actuales, debido a la naturaleza misma del desarrollo de estas tecnologías.

Por lo tanto, podemos decir que Cloud firestore nos ofrece ventajas que el SQL tradicional no puede ofrecernos y debido a esto, nos es más útil utilizar estas tecnologías en nuestro desarrollo de aplicaciones de React Native.

“Este estudio ha analizado minuciosamente las dos bases de datos NoSQL que ofrece Firebase: Cloud Firestore y Realtime Database. Se examinaron las características, los beneficios y las diferencias de cada uno al compararlos con las bases de datos SQL tradicionales.

El análisis concluyó que Cloud Firestore es la mejor opción para la mayoría de las aplicaciones React Native actuales.