

# Nombre del trabajo:

# Primera Fase Proyecto

# Materia:

# Diseño y Programación de Software Multiplataforma DPS941 G01T

# Alumno:

|  |  |  |
| --- | --- | --- |
| Apellidos | Nombres | Carné |
| Molina | Daniel Ernesto | MM211719 |
| Pérez Molina | Edgardo Baltazar | PM211493 |

# Docente: Lic. Alexander Alberto Sigüenza Campos

# Fecha de entrega:

07/05/2023

**¿Diferencias de base datos SQL y NoSQL?**

Una base de datos SQL vs NoSQL (Not Only SQL) cumple la misma labor en la administración y análisis de datos, sin embargo, una base de datos relacional o SQL trabaja con datos estructurados, mientras que una NoSQL puede manipular los datos no estructurados y otros tipos de datos.

De manera que, su mayor diferencia radica en la consistencia y tipo de información que maneja. La base de datos SQL se especializa en una cantidad y tipo específico de datos al ser una base de datos relacional y organizar de forma estructurada la información por medio de filas y columnas. En suma, se abastece de una serie de identificadores o claves mediante el lenguaje SQL.

Por otra parte, la base de datos NoSQL se emplea mucho más cuando el volumen de datos es mayor, puesto que no distingue los formatos de los datos, de manera que los puedes mezclar en documentos como PDF, Word, Excel, SMS, audio, vídeo, emails, etc.

**¿Las diferencias entre Firestore y Realtime?**

Realtime Database es la primera base de datos que fue liberada por Firebase con una alta eficiencia, baja latencia y capacidades para consultas en tiempo real. Por otra parte, Firestore es la nueva base de datos diseñada y desarrollada por Google la cual tiene una estructura mucho más fácil de entender, diseñar y tiene muchas mejoras en los SDKs para la realización de consultas.

Si hacemos una comparación en cuanto a estructura de datos, Realtime Database almacena los datos en JSON, pero se organiza o se representa en la plataforma como un árbol. Esta forma es compleja cuando crecen las anidaciones, las jerarquías o la información, mientras que Firestore tiene una estructura visible en la consola más sencilla y fácil de escalar cuando se tiene grandes jerarquías con el manejo de subcolecciones. Realtime Database da soporte de almacenamiento offline para Android y iOS, sin embargo, Firestore da soporte offline a móviles y también tiene soporte para web.

**¿La mejor opción para implementar React Native?**

Cloud Firestore es la base de datos más reciente de Firebase que se utiliza para el desarrollo de software y aplicaciones para dispositivos móviles. Aprovecha lo mejor de Realtime Database con un nuevo modelo de datos y más intuitivo. Con Cloud Firestore también se pueden realizar consultas más extensas, exactas y rápidas, y el escalamiento de forma automática, no requiere que comparta los datos a través múltiples instancias y se ajusta a un nivel más alto que Realtime Database para soportar consultas más complejas y puede combinar filtros con ordenamiento. Al realizar la consulta solo trae el documento o documentos sin llamar a subcolecciones o hijos, esto se puede pedir a demanda, por otro lado, las consultas son indexadas por defecto, también el ordenamiento puede ser descendente.

En Firestore el tipo de cobro es más que todo por operaciones de escritura, lectura y eliminado, el modelo secundario es por el tamaño de almacenamiento, el ancho de banda utilizado y los índices creados., volviéndose muy económico.

Realtime Database es la base de datos original producida por Firebase. Es una solución eficiente y de baja latencia destinada a las aplicaciones para dispositivos móviles que necesitan estados sincronizados entre ellos en tiempo real para ergonomía del usuario. Realtime Database soporta solo filtrar y ordenar un campo por un solo query. Además, cuando se realiza una consulta retorna todo el subárbol del nodo. El cobro por el almacenamiento en Realtime Database es por tamaño del volumen de datos y por el ancho de banda, es por esto que es más costoso que Firestore.

**Conclusiones**

Después de realizar nuestra investigación sobre las bases de datos SQL Y NoSQL (Not Only SQL) llegamos a la conclusión que cada una tiene sus ventajas a la hora de implementar una base de datos para el desarrollo de un nuevo software o una nueva aplicación, en SQL nos permite guardar información de una forma lineal por columnas y filas en las cuales nos permitan guardar datos de forma ordenada y que se puede acceder a esta o guardar nueva mediante Querys o líneas de comando que se utilizan para interactuar con la base de datos pero la cantidad información es limitada en varios factores entre los que se incluye el Hardware y el volumen de la base de datos, lo cual a su debido tiempo necesitaría de una inversión para el mejoramiento del almacenaje de la base de datos, por otro lado si utilizamos NoSQL (Not Only SQL), el cual podríamos utilizar atreves de herramientas como Firestore con el cual podríamos almacenar grandes cantidades de información y otro tipo de documentos los cuales podemos listar y relacionar, este tipo de almacenamiento se esta utilizando para la creación de aplicaciones y el desarrollo de nuevo software que permiten desplegar información en tiempo real para que el usuario tenga acceso a su información de manera mas sencilla y tener acceso a información muy práctica, es por esto que como grupo nos inclinamos mas por el uso de herramientas como Firestore que permite almacenar grandes volúmenes de información a un costo razonable y que es al sobrepasar ciertos limites que se nos empezaría a cobrar por lo que hemos llegado a la conclusión que ambos métodos tienen sus ventajas en cuanto a almacenaje y costo pero en estos momento Firestore sería más económico.