

**UNIVERSIDAD MARIANO GALVEZ DE GUATEMALA**  
**FACULTAD DE INGENIERIA EN SISTEMAS, SECCIÓN "B"**  
**CURSO: Introducción a los Sistemas de Cómputo**  
**Ingeniero Marco Tulio Valdez Albizurez**



**Bases de datos Relacionales y no Relacionales**

**Estudiante**

**Evelyn Guadalupe Jacobo Reynoso**

**0907-24-21711**

**Jalapa, 13 de abril de 2024.**

## **Introducción.**

Las bases de datos no relacionales y relacionales son factores de alta influencia respecto a la calidad del proyecto, aunque una es más antigua que la otra que en este caso es la base de dato relacional no quiere decir que sea de carácter antiguo, sino al contrario hoy en día se sigue utilizando y sigue siendo más popular ya que su elemental colección de elementos con conjugaciones de tablas formales descriptivas en el cual se puede acceder de muchas maneras sin tener que reorganizar las tablas de la base, es por ello que se la información se parte mediante identificadores, además son más robustas para almacenamiento ante cualquier falla. Por otro lado, se obtiene los datos no relacionales que usan variedades de modelos de datos, que incluyen documentos, gráficos, claves o valor, memorias y búsqueda, de igual forma los datos crecen muy rápido en momentos puntuales donde toda necesidad del proceso obtiene usos del sistema por parte de los usuarios múltiples ocasiones. Para las bases de datos relacionales y no relacionales es importante saber la importancia que imparte cada una de ellas tales como sus ventajas y desventajas, sus caracterizaciones, MongoDB, RavenDB, CouchDB, Firebase, y todos los formatos de datos utilizado por los motores para obtener una sistematización más estructurada a la hora de realizar cualquier formato ejecutado.

## **Bases de datos no relacionales**

Las bases de datos no relacionales son un sistema de almacenamiento de información que se caracteriza por no usar el lenguaje SQL para las consultas. Esto no significa que no puedan usar el lenguaje SQL, pero no lo hacen como herramienta de consulta, sino como apoyo. Por ello también se les suele llamar NoSQL o no solo SQL, de igual forma trabajan con estructuras definidas. Es decir, los datos no se almacenan en tablas, y la información tampoco se organiza en registros o campos. Tienen una gran escalabilidad y están pensadas para la gestión de grandes volúmenes de datos. Por otro lado, a diferencia de las bases de datos relacionales no cumple con el estándar ACID de atomicidad, consistencia, aislamiento y durabilidad. Las bases de datos no relacionales son más actuales que las relacionales, y su desarrollo se ha basado en la necesidad de crear sistemas de gestión capaces de trabajar con datos no estructurados o semiestructurados.

Principalmente, las bases de datos no relacionales se emplean para almacenar datos no estructurados, o de los que no se tienen ningún esquema o información predefinida. Sin embargo, también pueden ser empleados para el almacenamiento y gestión de datos estructurados una base de datos no relacional no requiere de tablas para el almacenamiento de la información, normalmente la estructura de una base de datos relacional se basa la organización de la información a través de documentos. Este tipo de databases están pensadas para ofrecer mayor escalabilidad horizontal y no tienen identificadores que permitan establecer relaciones entre diferentes conjuntos de datos de tal manera que cada uno de los documentos almacenados en la base de datos incluye todos los atributos del elemento, por lo que resultan muy útiles a la hora de guardar información poco estructurada o de la que no se tiene un esquema claro de inicio, de último las bases de datos no estructurales son cada vez más utilizadas ya que permiten almacenar datos de los cuáles no se tiene una estructura clara y que serían casi imposibles de organizar en una base relacional.

### **Ventajas:**

- ✓ Son mucho más flexibles a la hora de crear esquemas de información, lo que las convierte en una solución ideal para el almacenamiento y gestión de datos no estructurados o semiestructurados.
- ✓ Ofrecen una mayor escalabilidad. Pueden soportar mayores volúmenes de datos y añadir mayor capacidad añadiendo nuevos módulos de software, sin necesidad de añadir nuevos servidores.
- ✓ Garantizan un alto rendimiento, ya que están diseñadas para trabajar con modelos de datos concretos y patrones de acceso específicos.
- ✓ Son muy funcionales, ya que cuentan con API exclusivas y proporcionan modelos de datos para trabajar con cada tipo de datos presentes en la base.

### **Desventajas:**

- ✓ No cumplen igual que las relacionales con las propiedades de atomicidad, consistencia, integridad y durabilidad.
- ✓ No son compatibles con determinadas consultas en lenguaje SQL.
- ✓ Carecen de un sistema estandarizado ya que todavía son bases de datos relativamente nuevas.
- ✓ Muchos sistemas de gestión de bases de datos relacionales son de código abierto y tienen una gran comunidad detrás programando soluciones y nuevas funcionalidades. En el caso de las bases de datos no relacionales este soporte es mucho más limitado.

**MongoDB:** es una base de datos orientada a documentos que pertenece a la familia de bases de datos NoSQL. A diferencia de las bases de datos relacionales típicas, MongoDB almacena los datos en documentos BSON (Binary JSON), que son documentos flexibles similares a JSON. Proporciona un modelo de datos sin esquemas, lo que facilita el trabajo con datos no estructurados o semiestructurados. La adaptabilidad de Mongo lo hace ideal para aplicaciones con requisitos en constante cambio o arquitecturas de datos en expansión.

Las bases de datos relacionales típicas, MongoDB presenta varias ventajas. Para empezar, su arquitectura de datos adaptable permite a los desarrolladores repetir y alterar fácilmente las estructuras de datos a medida que el programa madura. También permite manejar sin problemas datos jerárquicos complejos. Además, la escalabilidad horizontal y la arquitectura distribuida de Mongo lo hacen ideal para tratar enormes cantidades de datos y cargas de tráfico significativas. Además, la capacidad integrada de replicación y conmutación por error automatizada de Mongo garantiza una alta disponibilidad y redundancia de datos.

Su función es que una arquitectura cliente o servidor. Los controladores, que proporcionan una API sencilla e intuitiva para las operaciones de base de datos, son utilizados por las aplicaciones para interactuar con MongoDB. las colecciones, que son equivalentes a las tablas de una base de datos relacional, son la forma en que Mongo guarda los datos. Cada colección está formada por varios documentos, que son análogos a las filas o registros de las bases de datos tradicionales, estos documentos, que pueden tener una gran variedad de estructuras, se guardan en formato BSON. MongoDB proporciona un completo lenguaje de consulta para recuperar y modificar datos.

**RavenDB:** Raven es un proyecto de código abierto, pero con una opción comercial para base de datos para plataformas NET/Windows. Raven ofrece un diseño de modelo de datos flexible para adaptarse a las necesidades de los sistemas del

mundo real, este sistema permite utilizar las consultas Linq que se basan en la baja latencia y el obtienen un alto rendimiento, Raven está publicado bajo licencia de código abierto y una licencia comercial, lo que significa que está libremente disponible, pero si se quiere utilizar con un software propietario, habría que adquirir una licencia comercial. Los documentos en Raven son esquemas JSON. Los documentos son entidades independientes no relacionadas entre sí. Sus características son Infraestructura escalable y la configuración sencilla en Windows, es fácil de instalar y se ejecuta en ventanas como un servicio normal o un sitio web de IIS7, de igual forma define fácilmente los índices con consultas LINQ, por último, API .NET. Raven viene con un api que implementa la unidad de trabajo.

**CouchDB:** Es una base de datos de documentos NoSQL de código abierto que recopila y almacena datos en formatos de documento basados en JSON. A diferencia de las bases de datos relacionales, CouchDB utiliza un modelo de datos sin esquema, lo que simplifica la gestión de registros en diversos dispositivos informáticos, teléfonos móviles y navegadores web, Para la mayoría de las empresas, decidir qué proveedor utilizar para suministrar la tecnología de gestión de datos puede ser complicado. El software propietario no solo impone ciertas restricciones de licencia sobre el uso de la tecnología, sino que también existen preocupaciones sobre la continuidad del negocio al migrar todos sus datos empresariales a un sistema de gestión de bases de datos de "talla única", sin visibilidad sobre su estructura interna.

Algunas características clave de CouchDB y sus diferencias con otras bases de datos NoSQL pueden ser que no existe distinciones si los datos están alojados en un servidor o en varios. Más bien, CouchDB identifica los cambios en los documentos a medida que ocurren desde cualquier fuente y garantiza que todas las copias de seguridad permanezcan sincronizadas con la información más actualizada. Esto permite la autocontención y la gestión de varias réplicas de base de datos, sin dejar de alojar información precisa y en tiempo real en múltiples entornos informáticos, así también usa vistas como herramienta principal para ejecutar consultas y crear informes a partir de archivos de documentos almacenados y es extremadamente accesible y ofrece una serie de ventajas de compatibilidad cuando se integra con su infraestructura actual.

**Firebase:** Es una plataforma digital que se utiliza para facilitar el desarrollo de aplicaciones web o móviles de una forma efectiva, rápida y sencilla, la cual es utilizada por sus diversas funciones como una técnica de Marketing Digital para aumentar la base de usuarios y generar mayores beneficios económicos, su principal objetivo, es mejorar el rendimiento de las apps mediante la implementación de diversas funcionalidades que van a hacer de la aplicación en cuestión, mucho más manejable, segura y de fácil acceso para los usuarios.

Sus principales características es que son herramientas soportadas en múltiples plataformas en el cual Firebase, se encuentra disponible para las principales

plataformas móviles, ya sea Android, IOS o la Web. De igual forma es caracterizada por permitir ganar dinero donde mediante el uso habilita la posibilidad de generar dinero mediante anuncios y publicidades, que permite un desarrollo gratuito de las apps. Esta plataforma digital, no pide pagar para la iniciación y uso de la misma ya que empieza a cobrar a la hora de manejarla en un nivel más avanzado y experimentado, puedes solicitar planes de mayor jerarquía según el uso que se le dé. Así mismo genera crecimiento y desarrollo en las aplicaciones esta es una de las características más importantes, puesto que se centra en el objetivo primordial de todo lo que es Firebase. Mediante los servicios ofrecidos, permite que puedas desarrollar de una forma sencilla, segura y rápida tus aplicaciones en el dispositivo que te encuentres.

### **Cuadro comparativo Bases de datos relacionales y no relacionales:**

<b>Bases de Datos Relacionales</b>	<b>Bases de Datos no Relacionales</b>
<ul style="list-style-type: none"> <li>❖ Se caracterizan por organizar la información en partes pequeñas, con identificadores a partir de los cuales esas partes pequeñas se relacionen entre sí.</li> <li>❖ Se organizan normalmente en tablas, no en documentos, como en el caso de las otras.</li> <li>❖ Se trata de bases de datos más robustas y con menos vulnerabilidades ante fallos</li> <li>❖ Garantizan el cumplimiento de las cualidades ACID: atomicidad, consistencia, integridad y durabilidad.</li> <li>❖ MySQL junto con Oracle, son las bases de datos de este tipo más conocidas. Luego siguen en la lista SQL Server y PostgreSQL.</li> <li>❖ Datan de la década del 80, que fue cuando comenzaron a utilizarse, pero, no significa que son antiguas o que pertenecen al pasado.</li> <li>❖</li> </ul>	<ul style="list-style-type: none"> <li>❖ Se trata de bases de datos que están diseñadas para soportar grandes volúmenes de datos, por lo que son bases de datos altamente escalables.</li> <li>❖ La información se suele almacenar en documentos y no en tablas, como en el caso de las bases de datos relacionales.</li> <li>❖ No garantizan el cumplimiento de atomicidad, consistencia, integridad y durabilidad, es decir, las cualidades ACID, que sí garantizan las bases de datos relacionales.</li> <li>❖ Se trata de bases de datos de gran utilidad.</li> <li>❖ No poseen identificadores que sirvan de relación entre un conjunto de datos y otros</li> <li>❖ No cuenta aún con un sistema estandarizado, ya que se trata de un sistema de almacenamiento de datos bastante nuevo.</li> <li>❖ MongoDB es la base de datos de este tipo sin dudas más conocida. Atrás suyo vienen Redis, Elasticsearch y Cassandra.</li> </ul>

## **Formatos de datos utilizado por los motores investigados:**

### **MongoDB:**

1. Documentos. En MongoDB la información se almacena en documentos, agrupados en colecciones que pertenecen a bases de datos donde los documentos de una colección no tienen por qué compartir la misma estructura, aunque en general sí que lo harán. Pero no es obligatorio.
2. JSON. En MongoDB se utiliza otro formato para almacenar los documentos, BSON, que no es más que una extensión de JSON que se representa de manera binaria. JSON (JavaScript Object Notation) es un formato estándar utilizado principalmente para transmitir datos entre un servidor web y una aplicación web. Inicialmente extraído de JavaScript, actualmente es independiente del lenguaje de programación.
3. BSON. Si MongoDB utilizase JSON para almacenar la información estaríamos limitados a seis tipos de datos, los cuales proporcionan tipos de datos que no existen en JSON, como Date para fechas, BinData para información binaria, ObjectId para valores únicos (generalmente usado para el campo \_id de las colecciones).
4. GridFS. En MongoDB, un documento BSON no puede ser superior a 16 MB. Así que independientemente del tipo de datos que estemos utilizando, para documentos grandes tenemos que utilizar GridFS, que básicamente lo que hace es dividir los documentos en trozos iguales (256 KBytes), y guardar cada uno de estos trozos en un documento.

### **RavenDB:**

1. Raven-Entity-Name es esta propiedad la que nos permite indicar que un documento se categoriza en una determinada colección.
2. Recurso dedicado a los attachments, al igual que los índices.
3. Raven es RESTfull y por lo tanto podremos mediante la identificación de recursos y verbos.
4. HTTP acceder a los documentos almacenados.

### **CouchDB:**

1. Base de datos documental sin schema. Estamos acostumbrados a que cuando hablamos de base de datos pensemos en el modelo relacional (columna, filas, tablas, relaciones) En cambio, CouchDB nos ofrece guardar nuestros datos de otra forma.

2. CouchDB no nos ofrece un lenguaje tipo SQL para realizar consultas, sino que nos ofrece un sistema basado en MapReduce para poder obtener los datos que queramos.
3. Accesible por REST nos permite acceder a nuestros datos de una forma muy sencilla a través de URLs.
4. Replicación integrada. Una funcionalidad relativamente exótica que nos permite que nuestra BD de datos sincronice sus datos de una forma muy sencilla (una simple llamada REST la activa) con otra BD remota o local. De este modo podemos tener de una forma sencillísima una o mas réplicas de nuestra BD para implementar arquitecturas de alta disponibilidad o de balanceo de carga.

#### **Firebase.**

1. Cloud Storage permite almacenar y descargar archivos como imágenes o vídeos.
2. Cloud Firestore es una base de datos flexible y escalable.
3. Firebase Realtime Database almacena datos de aplicación JSON.
4. Firebase Remote Config almacena pares clave-valor.



### **Conclusión:**

Es importante tener conocimientos determinados ante estas bases de datos ya que organizan todo tipo de necesidades de proceso para todo tipo de volumen que crezca muy rápido o crece de forma lenta, son modelos diferentes que ofrecen ventajas y desventajas cada una, sin embargo, solucionan de manera eficaz los problemas presentados que posee cada base de datos, cada una cuenta con conformidades por documentos, campos y datos del que se puedan almacenar las ejecuciones, para conllevar todo de manera sistemática y organizada se debe saber como manipular cada dato desde la primera estructura. Si el desarrollador de software se encuentra en una situación de elegir entre cada uno de ellas no se debe de observar como un dilema dependiente sino en realidad con las dos opciones es posible constituir los sistemas dependiendo de lo que se desee. Como ya se observó anteriormente cada uno imparte con ventajas y desventajas que son importantes observar cuales son cada uno de ellas para tener un conocimiento claro y preciso de ellos.

Por ellos existe conjugaciones importantes que son los MongoDB es una base de datos orientada a documentos que pertenece a la familia de bases de datos NoSQL, RavenDB es un proyecto de código abierto, pero con una opción comercial para base de datos para plataformas NET/Windows, CouchDB es una base de datos de documentos NoSQL de código abierto que recopila y almacena datos en formatos de documento basados en JSON, por último se obtiene Firebase es una plataforma digital que se utiliza para facilitar el desarrollo de aplicaciones web o móviles de una forma efectiva, rápida y sencilla

### **E grafías:**

- <https://ayudaleyprotecciondatos.es/bases-de-datos/no-relacional/>
- <https://aprendeinformaticas.com/mongodb-que-es/>
- [https://programacion.net/noticia/raven-db\\_1703](https://programacion.net/noticia/raven-db_1703)
- <https://www.ibm.com/es-es/topics/couchdb>
- <https://rockcontent.com/es/blog/que-es-firebase/>
- <https://www.obsbusiness.school/blog/que-diferencia-una-base-de-datos-relacional-vs-una-no-relacional>
- <https://www.pragma.com.co/academia/lecciones/bases-de-datos-relacionales-vs.-no-relacionales>
- <https://aula301.com/tipos-datos-podemos-utilizar-mongodb/>
- <https://geeks.ms/unai/2011/12/05/ravendb-ii-los-documentos/>
- <https://www.rgnu.com.ar/2010/03/30/apache-couchdb-una-base-de-datos-nosql-relax/>
- <https://www.paradigmadigital.com/dev/que-es-firebase-cloud-storage/>