

NoSQL: Una alternativa de solución para grandes volúmenes de información

José Juan Ulín Ricárdez, Miguel Pérez Vasconcelos, Rosa Gómez Domínguez, Fidelio Castillo Romero

Tecnológico Nacional de México Campus Villahermosa

Resumen

Las bases de datos NoSQL han surgido ante la necesidad de responder a las altas demandas de procesamiento de grandes volúmenes de datos en ambientes distribuidos, escenarios donde las nuevas aplicaciones han transitado a entornos colaborativos con una mayor interacción de los usuarios provocados por el fenómeno de las redes sociales, el comercio electrónico, inteligencia artificial e Internet de las cosas (IoT). En este artículo, se analizan y comparan las principales características de las bases de datos NoSQL en relación a las bases de datos relacionales, evaluando las ventajas e inconvenientes de ambas opciones. También se identifican los principales retos y desafíos a los que se enfrentan las empresas al trabajar con grandes cantidades de datos, y cómo las bases de datos NoSQL pueden ayudar a superar estos desafíos.

Abstract

The NoSQL databases have emerged in response to the need to handle high processing demands of large volumes of data in distributed environments, scenarios where new applications have transitioned to collaborative environments with greater user interaction caused by the phenomena of social networks, e-commerce, artificial intelligence, and the Internet of Things (IoT). This article analyzes and compares the main characteristics of NoSQL databases in relation to relational databases, evaluating the advantages and disadvantages of both options. It also identifies the main challenges and challenges faced by companies when working with large amounts of data, and how NoSQL databases can help overcome these challenges.

Palabras Clave: NoSQL, SGBD, Escalabilidad, Disponibilidad, Particionamiento

Keywords: NoSQL, DBMS, Scalability, Availability, Partitioning

1. INTRODUCCIÓN

En la actualidad, las empresas enfrentan el desafío de trabajar con grandes cantidades de datos de manera eficiente. El uso de bases de datos tradicionales, como las bases de datos relacionales, puede ser limitado y costoso a medida que la cantidad de datos aumenta. En este contexto, han surgido las bases de datos NoSQL como una alternativa atractiva para manejar grandes volúmenes de datos. Las bases de datos NoSQL se han convertido en una tecnología de amplia aceptación en el campo de la gestión de datos debido a su capacidad para manejar datos no estructurados y semiestructurados, su escalabilidad y su capacidad de procesamiento de alta velocidad.

En este artículo se proponen sistemas de gestión de bases de datos no relacionales para resolver las limitaciones de los SGBD tradicionales. Las bases de datos NoSQL son libres de esquemas; están diseñadas para trabajar de forma flexible con los datos no estructurados.

2. RETOS Y DESAFÍOS AL TRABAJAR CON GRANDES VOLÚMENES DE DATOS

En la actualidad, las empresas se enfrentan a una cantidad cada vez mayor de datos que deben ser almacenados, procesados y analizados de manera efectiva para obtener información valiosa. Sin embargo, este

aumento en la cantidad de datos también conlleva una serie de retos y desafíos que deben ser abordados para aprovechar al máximo su potencial (Chen & Zhang, 2014).

La razón por la cual los SGBD relacionales no son adecuados para satisfacer los requisitos de las aplicaciones modernas como las orientadas a redes sociales, big data e IoT, es su completa dependencia de la definición de esquemas predefinidos fijos, reglas de consistencia de las transacciones, procesamiento de consulta mediante la unión de varias tablas, lo cual repercute en el rendimiento y desempeño cuando se trata de grandes volúmenes de datos y que además se almacenan de manera distribuida en múltiples servidores, un rendimiento deficiente en términos de disponibilidad y escalabilidad al manejar grandes volúmenes de cargas de trabajo no estructuradas o semiestructuradas.

En los escenarios donde se realizan consultas complejas esto puede traducirse en un alto consumo de procesamiento lo que repercute de manera directa en el tiempo de respuesta, degradando de esta manera el desempeño de la base de datos, lo que puede limitar la capacidad de las empresas para obtener información útil en tiempo real y de manera eficaz.

En relación a la escalabilidad, las bases de datos relacionales suelen tener dificultades para escalar horizontalmente, es decir, agregar más nodos o servidores para atender la demanda de un mayor volumen de datos, y un mayor número de usuarios concurrentes. Este problema puede generar cuellos de botella en el rendimiento y desempeño de la base de datos, un aumento de los costos de infraestructura, y capacidad limitada para atender los nuevos requerimientos de la organización que satisfagan las necesidades de los usuarios.

La consistencia de los datos es otro desafío importante. Las bases de datos relacionales garantizan la consistencia de los datos utilizando transacciones ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad), pero en consecuencia con un rendimiento más lento. Por otro lado, las bases de datos NoSQL suelen tener modelos de consistencia más flexibles, pero esto puede llevar a problemas de integridad de los datos (Kunda & Phiri, 2017).

3. COMO LAS BASES DE DATOS NOSQL AYUDAN A SUPERAR LOS DESAFÍOS DE LOS SGBD RELACIONALES

Las bases de datos NoSQL han surgido como una alternativa a las bases de datos relacionales para manejar grandes cantidades de datos y superar los desafíos que plantean. A continuación, se describen algunas de las formas en que las bases de datos NoSQL pueden ayudar a superar estos desafíos.

Escalabilidad horizontal

Una de las ventajas de las bases de datos NoSQL es su capacidad para escalar horizontalmente. Esto significa que pueden manejar grandes volúmenes de datos al distribuirlos en múltiples servidores, lo que permite un mejor rendimiento y una mayor capacidad de procesamiento (Kunda & Phiri, 2017). Una de las principales razones por la que muchas empresas están implementando bases de datos NoSQL tiene que ver con la escalabilidad horizontal, ya que ello le permite adaptarse de forma dinámica a los nuevos requerimientos en el incremento del número de usuarios concurrentes y altas prestaciones en capacidades de cómputo.

Flexibilidad en el esquema de datos

Las bases de datos NoSQL son conocidas por su flexibilidad en el esquema de datos, lo que significa que no tienen una estructura fija como las bases de datos relacionales. Esto permite una mayor facilidad para adaptarse a diferentes tipos de datos, y a menudo se utilizan en aplicaciones que requieren una gran cantidad de datos no estructurados. Esto quiere decir que ante la necesidad de agregar nuevos datos y la modificación de los existentes no hay necesidad de cambiar la estructura de la base de datos.

Alta disponibilidad y tolerancia a fallos

Las bases de datos NoSQL también son conocidas por su alta disponibilidad y tolerancia a fallos. En lugar de tener un único servidor centralizado que puede fallar, las bases de datos NoSQL se distribuyen en múltiples servidores y tienen sistemas de replicación incorporados que permiten una mayor tolerancia a fallos y una mayor disponibilidad de los datos (Erl et al., 2016). Por lo que la replicación de bases de datos en NoSQL es una de las principales ventajas para poder ofrecer alta disponibilidad y tolerancia a fallos.

Mejor rendimiento en consultas complejas

Las bases de datos NoSQL a menudo tienen un mejor rendimiento que las bases de datos relacionales cuando se trata de consultas complejas en grandes cantidades de datos. Esto se debe en parte a su capacidad para realizar operaciones de lectura/escritura en paralelo (Sadalage & Fowler, 2012) y a la optimización de consultas en tiempo real (de Oliveira et al., 2021).

4. BASES DE DATOS RELACIONALES

Los Sistemas Gestores de Bases de Datos (SGBD) también conocidos como DBMS (Data Base Manager System) son un tipo de software que consiste en un conjunto de programas cuya función principal consiste en almacenar y gestionar un conjunto de datos interrelacionados entre sí. Estos sistemas están basados en el modelo relacional que hace referencia a un conjunto de tablas mediante las cuales es posible representar los datos y las relaciones entre ellos, cada tabla está compuesta por un conjunto de tuplas o filas denominada relación, es decir una tabla se puede concebir como un conjunto de relaciones. Ahora bien, para poder realizar las distintas operaciones en un modelo relacional, los SGBD utilizan el lenguaje estructurado de consultas SQL como su lenguaje de interfaz de programación de aplicaciones, tanto para la definición de los esquemas de la base de datos como para las operaciones generales que se realiza con los datos (Elmasri & Navathe, 2016).

En el mercado de las bases de datos se utiliza con mucha frecuencia el término SGBD para referirse a un producto específico de software destinado a la gestión de los datos, siendo los más utilizados: MySQL, PostgreSQL, SQLite, Microsoft SQL Server, Firebase, Oracle y DB2 (Stack Overflow Developer Survey, 2021).

La experiencia de diversos tipos de organizaciones en la implementación de los SGBD basados en el modelo relacional para la representación y almacenamiento de datos estructurados mediante esquemas, así como la recuperación y procesamiento de los datos para generar nueva información, ha demostrado ser una solución óptima para la creación de transacciones de forma segura, más sin embargo cuando se trata de almacenar y procesar grandes volúmenes de datos heterogéneos con niveles de desempeño aceptables, se genera un

efecto perjudicial en su rendimiento, escalabilidad y flexibilidad, y es ahí donde surge el modelo de base de datos NoSQL para dar respuesta a estos nuevos requerimientos.

5. BASES DE DATOS NOSQL

Las Bases de Datos NoSQL, también conocidas como No Only SQL (No solo SQL) o No Relational, surgieron como una alternativa a las tradicionales Bases de Datos Relacionales (BDR) que no podían cumplir con las demandas de rendimiento, escalabilidad, y disponibilidad de las aplicaciones modernas (Durán-Cazar et al., 2019; Gupta et al., 2018). Estas bases de datos se caracterizan por no utilizar un esquema fijo para definir la estructura de los datos (Marrero et al., 2019), lo que permite una mayor flexibilidad en el manejo de grandes volúmenes de información no estructurada o semiestructurada (Grolinger et al., 2014).

Los SGBD para garantizar el procesamiento de transacciones incorporan en su modelo las propiedades ACID (Atomicity, Consistency, Isolation, Durability), mientras que las bases de datos NoSQL siguen la semántica BASE (Erl et al., 2016):

BA (Basically Available) Básicamente disponible: una aplicación está lista para aceptar solicitudes de lectura/escritura todo el tiempo.

S (Soft state) Estado blando: Es posible que los resultados no siempre se basen en datos coherentes (no hay garantía de coherencia).

E (Eventual consistency) Coherencia eventual: el sistema asegura que los datos serán consistentes en algún momento posterior.

Taxonomía de bases de datos NoSQL

Existen diferentes tipos de bases de datos NoSQL, cada una con sus propias características y funcionalidades. Las principales categorías de bases de datos NoSQL incluyen: Documentales, Columnares, Clave-Valor y Grafos (Gupta et al., 2018; Makris et al., 2016; Matallah et al., 2021).

Bases de datos documentales. Los datos se almacenan en documentos basados en el formato JSON y que pueden contener cualquier número de campos y anidaciones.

Bases de datos columnares. Los datos se almacenan por columnas en lugar de por filas, lo que permite una mayor eficiencia en la consulta de datos específicos.

Bases de datos clave-valor. Los datos se almacenan en un diccionario de claves y valores lo que las hace ideales para aplicaciones de caché y almacenamiento en memoria.

Bases de datos de grafos. Los datos se almacenan como nodos y aristas, lo que facilita la representación y consulta de relaciones complejas entre los datos.

Cada tipo de base de datos NoSQL tiene sus propias ventajas y desventajas en términos de rendimiento, escalabilidad y facilidad de uso. Por lo tanto, es importante seleccionar la base de datos NoSQL adecuada para

cada aplicación en función de sus necesidades específicas. En general, las bases de datos NoSQL son adecuadas para aplicaciones que requieren alta disponibilidad, escalabilidad horizontal, baja latencia y manejo eficiente de grandes cantidades de datos no estructurados o semiestructurados.

De acuerdo a los datos publicados en Stack Overflow Developer Survey (2021), los sistemas gestores de bases NoSQL más utilizados por los desarrolladores para sus proyectos de software son: MongoDB, Redis, Elasticsearch, DynamoDB, Cassandra, y Couchbase.

6. BASES DE DATOS NOSQL: VENTAJAS Y LIMITACIONES

Derivado de la revisión de la literatura, se han identificado las principales ventajas que proporcionan las bases de datos NoSQL.

Volumen y variedad de datos

La transición de la web 2.0 a la web 3.0 donde los procesos se desarrollan en entornos colaborativos, se caracteriza por el gran volumen de información que se genera, así como la gran variedad de datos. Los sistemas de bases de datos relacionales no han podido responder de manera eficiente en el manejo de estos grandes volúmenes de datos provenientes de diversas fuentes. Es precisamente este escenario donde sobresale el modelo NoSQL para el manejo de grandes volúmenes de datos (Chen & Zhang, 2014), y para las aplicaciones de Internet intensivas en datos, como Google, Facebook, Instagram, que han migrado a NoSQL.

Escalabilidad

Las bases de datos NoSQL proporcionan escalabilidad horizontal, y consiste en la capacidad para satisfacer de forma eficiente las necesidades de cargas de trabajo variables en términos de recursos y rendimiento (Kunda & Phiri, 2017). Una creciente demanda originada por el número de usuarios requiere agregar más servidores, de forma contraria conforme la demanda disminuye y el tráfico se normaliza, se liberan servidores adicionales.

Flexibilidad

Una de las características sobresalientes del esquema de bases de datos no relacional o NoSQL, consiste en que no posee un esquema o definición de la base de datos que es fundamental en las bases de datos relacionales (Matallah et al., 2021); con lo cual se logra flexibilidad para soportar variaciones significativas en las estructuras de la base de datos.

Disponibilidad

Las bases de datos y aplicaciones NoSQL por lo general se implementan en sistemas distribuidos que consisten en un conjunto de servidores de bajo costo, cada uno con una réplica de la base de datos. En caso de ocurrir un fallo en el servidor principal, los servidores secundarios eligen un nuevo servidor principal permitiendo así que la base de datos esté siempre disponible (Makris et al., 2016). Por otra parte, si hubiese un aumento en el número de usuarios el cluster de servidores se expande en su capacidad de almacenamiento, capacidad de cómputo y ancho de banda para mantener el rendimiento de la base de datos.

Replicación

Uno de los sistemas gestores de bases NoSQL más populares es MongoDB, el cual utiliza un modelo de conjunto de réplicas que le permite distribuir copias de datos entre los servidores que conforman el clúster. Las operaciones de lecturas intensivas en la base de datos pueden distribuirse entre los nodos del clúster del conjunto de réplicas (Makris et al., 2016), donde cada escritura en el servidor principal se transmitirá a todos los servidores secundarios.

Costo

Los sistemas gestores de bases de datos NoSQL como MongoDB, CouchDB y Cassandra son de código abierto (Kunda & Phiri, 2017), por lo cual se evitan problemas de licenciamiento que son muy comunes y costosos en los sistemas tradicionales de bases de datos relacionales como Oracle y Microsoft SQL Server. Esto brinda la oportunidad a investigadores que necesitan realizar pruebas de las características de una base de datos, y proporciona un almacenamiento más económico a los usuarios que no pueden pagar un licenciamiento de software propietario.

Particionamiento (Sharding)

Otra de las grandes ventajas de las bases de datos NoSQL en relación a las bases de datos relacionales es el particionamiento de datos en múltiples servidores para garantizar el rendimiento y disminuir el tiempo de latencia. Para lograr esto, se particiona el conjunto de datos en varios servidores (Erl et al., 2016; Makris et al., 2016) y se replica cada porción en muchos servidores, aumentando significativamente la velocidad de lectura y escritura del sistema ya que diferentes usuarios acceden a distintos fragmentos del conjunto de datos.

Limitaciones y desventajas de las bases de datos NoSQL

Al iniciar un proyecto donde se integre el modelo NoSQL como la arquitectura de los datos es muy importante tener en cuenta sus limitaciones y desventajas:

Ausencia de un estándar. Una de las desventajas de NoSQL es que carece de un lenguaje de consulta estándar (Matallah et al., 2021). Cada sistema gestor NoSQL incorpora su propio lenguaje e interfaz, lo cual ha dificultado la aceptación amplia de NoSQL.

Ausencia de la propiedad ACID. Debido a la alta disponibilidad que ofrecen las bases de datos NoSQL, no ofrecen una consistencia en transacciones donde la integridad de los datos es fundamental. Por ejemplo, si se necesita una alta integridad de los datos, es posible que una base de datos NoSQL no sea la mejor opción, ya que no ofrecen transacciones atómicas y pueden ser menos consistentes que las bases de datos relacionales (Sadalage & Fowler, 2012).

Madurez del sistema gestor. Las bases de datos NoSQL son menos maduras que las bases de datos relacionales (Matallah et al., 2021), lo que significa que pueden tener menos características y pueden ser menos estables.

Seguridad. El esquema de seguridad de los datos no está suficientemente desarrollado, es decir no está implementado a nivel de esquema de la base de datos, y se depende en gran medida en la parte de la aplicación para la protección de datos (Kunda & Phiri, 2017).

Bases de Datos Relacionales y NoSQL

En la Tabla 1, se muestra un comparativo entre los SGBD relacionales y NoSQL.

Tabla 1. Comparación de los SGBD Relacionales y NoSQL. Elaboración propia.

Característica	SGBD Relacional	NoSQL
Modelo de Datos	Relacional	Documentos, Clave-Valor, Columnar, Grafos
Datos	Estructurados	Estructurados, Semi-estructurados, No Estructurados
Esquemas	Fijos	Dinámicos
Escalabilidad	Vertical	Horizontal
Manejo de transacciones	ACID	BASE
Arquitectura	Centralizada	Distribuida
Lenguaje de Consulta	SQL	No está estandarizado.
Desempeño en grandes volúmenes de datos.	Bajo	Alto
Tipo de Aplicación	Bancos, transacciones financieras	Redes sociales, IoT

7. CASO DE ÉXITO EN LA IMPLEMENTACIÓN DE UNA BASE DE DATOS NOSQL

Creando una ciudad más segura y más inteligente (Chicago Uses MongoDB To Create A Smarter And Safer City, n.d.).

En 2011, Brett Goldstein jefe de datos del departamento de policía de la ciudad de Chicago, construyó un prototipo que permitía analizar datos para mejorar la vigilancia, mejorar los servicios y disponer los datos de la ciudad al público. Inicialmente disponía de un solo nodo implementado sobre una base de datos NoSQL con MongoDB en el cual se vinculaba datos almacenados de forma aislada en los sistemas de la ciudad; ello permitía crear mapas que ofrecían diversas vistas en tiempo real sobre las que podían actuar de forma inmediata.

Una vez que se validó el prototipo, surgieron nuevos requerimientos: un nuevo enfoque de arquitectura de los datos, combinar de forma masiva datos estructurados y no estructurados, análisis de datos en tiempo real, e

implementación inmediata del nuevo sistema. Es así como se crea la aplicación Windy Grid convirtiéndose en una plataforma geoespacial en tiempo real que permite el almacenamiento flexible de documentos, el análisis rápido y potente, y la arquitectura escalable de MongoDB.

Mediante la implementación de MongoDB en la plataforma WindyGrid, se creó un sistema nervioso central que reúne diariamente siete millones de registros(documentos) de los diferentes departamentos, combinando de manera creativa el análisis de datos con mapas visuales lo que permite a los gerentes obtener información en tiempo real sobre las operaciones de la ciudad, información que nunca antes habían tenido.

Mediante la combinación de datos, mapas y análisis de WindyGrid se logra:

- Hacer análisis predictivo.
- Identificar problemas potenciales antes de que se conviertan en problemas mayores.
- Coordinar las acciones de respuestas entre los departamentos.
- Los gerentes pueden actuar en el momento, en lugar de después del hecho.
- Pueden anticipar, en lugar de simplemente reaccionar ante los problemas.

Con el esquema de datos flexible de MongoDB no fue necesario reemplazar los sistemas administrativos existentes, sino reunir los datos que genera cada aplicación en una nueva aplicación, lo que permite crear nuevos esquemas en tiempo real que se vuelve crucial a medida que los requerimientos de la plataforma Windy Grid se expanden y agregan cada día millones de registros de datos estructurados y no estructurados.

Esta agregación de datos flexible y escalable es la columna vertebral para ejecutar análisis predictivos en tiempo real. Es una colección en constante cambio y en constante crecimiento que rastrea lo que hace funcionar a la ciudad, o lo que no, actualizaciones de obras viales, demoras en la recolección de basura, emergencias de salud del 911, quejas sobre el ruido del 311, tweets públicos sobre el funcionamiento de la ciudad, ubicaciones de autobuses a lo largo de su ruta, patrones de semáforos y mucho más.

8. DISCUSIÓN Y CONCLUSIONES

Con el crecimiento exponencial de los datos, las bases de datos NoSQL han surgido como una alternativa muy importante al modelo de bases de datos relacionales. Es importante mencionar que las bases de datos relacionales, aunque tienen un grado de madurez y rendimiento en su uso durante ya varias décadas en aplicaciones que gestionan la información en bases de datos convencionales, su diseño no siempre es escalable para las enormes cantidades de datos que se manejan actualmente, así como tener un desempeño eficiente en términos de velocidad. En cambio, las bases de datos NoSQL pueden manejar grandes cantidades de datos y proporcionar una flexibilidad y escalabilidad excepcionales.

A través de esta investigación, se ha demostrado que las bases de datos NoSQL tienen una amplia variedad de características y herramientas para satisfacer las necesidades de cualquier organización que trabaje con grandes cantidades de datos. Sin embargo, también se ha identificado una serie de limitaciones y desventajas que las organizaciones deben considerar antes de elegir una base de datos NoSQL. Aunque estas limitaciones son importantes, no deben ser un obstáculo para la adopción de las bases de datos NoSQL, especialmente si se utilizan de manera correcta y estratégica.

Las bases de datos NoSQL son una alternativa valiosa y en constante evolución para las bases de datos relacionales. Si bien las bases de datos relacionales todavía pueden ser la mejor opción para algunas aplicaciones, las bases de datos NoSQL tienen un lugar importante en el mundo de las grandes bases de datos. Con su capacidad para manejar grandes cantidades de datos y proporcionar una flexibilidad y escalabilidad excepcionales, las bases de datos NoSQL seguirán siendo una opción atractiva para las organizaciones que necesitan manejar grandes cantidades de datos en el futuro.

REFERENCIAS

- [1] Chen, P., & Zhang, C.-Y. (2014). Data-intensive applications, challenges, techniques and technologies: A survey on Big Data. *Information Sciences*, 275, 314-347. <http://dx.doi.org/10.1016/j.ins.2014.01.015>
- [2] Chicago Uses MongoDB To Create A Smarter And Safer City. (n.d.). MongoDB. Retrieved March 5, 2023, from <https://www.mongodb.com/customers/city-of-chicago>
- [3] de Oliveira, V. F., de Oliveira Pessoa, M. A., Junqueira, F., & Miyagi, P. E. (2021). SQL and NoSQL Databases in the Context of Industry 4.0. *Machines*, 10(1). <http://dx.doi.org/10.3390/machines10010020>
- [4] Durán-Cazar, J. W., Tandazo-Gaona, E. J., Morales-Morales, M. R., & Morales Cardoso, S. (2019). Rendimiento de bases de datos columnares. *Ingenius. Revista de Ciencia y Tecnología*, 22, 47-58. <https://doi.org/10.17163/ings.n22.2019.05>
- [5] Elmasri, R., & Navathe, S. (2016). *Fundamentals of Database Systems* (7th ed.). Pearson.
- [6] Erl, T., Buhler, P., & Khattak, W. (2016). *Big Data Fundamentals: Concepts, Drivers & Techniques* (T. Erl, Ed.). Prentice Hall.
- [7] Grolinger, K., Hayes, M., Higashino, W. A., L'Heureux, A., Allison, D. S., & Capretz, M. A.M. (2014, September 22). Challenges for MapReduce in Big Data [2014 IEEE World Congress on Services]. IEEE. <https://ieeexplore.ieee.org/document/6903263>
- [8] Gupta, A., Tyagi, S., Panwar, N., Sachdeva, S., & Saxena, U. (2018). NoSQL databases: Critical analysis and comparison. 2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN). <https://doi.org/10.1109/IC3TSN.2017.8284494>
- [9] Kunda, D., & Phiri, H. (2017). A Comparative Study of NoSQL and Relational Database. *ZAMBIA INFORMATION COMMUNICATION TECHNOLOGY (ICT) JOURNAL*, 1(1), 1-4. https://www.researchgate.net/publication/326019759_A_Comparative_Study_of_NoSQL_and_Relational_Database
- [10] Makris, A., Tserpes, K., Andronikou, V., & Anagnostopoulos, D. (2016). A classification of NoSQL data stores based on key design characteristics. *Procedia Computer Science*, 97, 94-103. <https://doi.org/10.1016/j.procs.2016.08.284>
- [11] Marrero, L., Olsowy, V., Thomas, P., Delia, L., Tesone, F., Fernandez Sosa, J., & Pesado, P. (2019). Un estudio comparativo de bases de datos relacionales y bases de datos NoSQL [XXV Congreso Argentino de Ciencias de la Computación]. SEDICI. <http://sedici.unlp.edu.ar/handle/10915/91403>
- [12] Matallah, H., Belalem, G., & Bouamrane, K. (2021). Comparative Study Between the MySQL Relational Database and the MongoDB NoSQL Database. *International Journal of Software Science and Computational Intelligence*, 13(3), 38-63. <https://doi.org/10.4018/IJSSCI.2021070104>
- [13] Sadalage, P. J., & Fowler, M. (2012). *NoSQL distilled: a brief guide to the emerging world of polyglot persistence*. Addison-Wesley Professional. Retrieved March 27, 2023, from <https://bigdata-ir.com/wp-content/uploads/2017/04/NoSQL-Distilled.pdf>
- [14] Stack Overflow Developer Survey 2021. (n.d.). Stack Overflow Annual Developer Survey. Retrieved February 20, 2023, from <https://insights.stackoverflow.com/survey/2021>

Correo de autor de correspondencia: jose.ulnr@villahermosa.tecnm.mx