Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

June 2026. Vol. 5. No. 3; e-ISSN: 2808-4519

MySQL vs MongoDB on Small-Scale E-Commerce Query Workloads: A Systematic Review of Database Performance Optimization

Efriati Annisa Hakim[1*], Dinda Puji Lestari[2], Vanisa Amalia[3], Imam Prayogo Pujiono[4]

[1,2,3,4]*Faculty of Islamic Economics and Business, Digital Business, UIN K.H. Abdurrahman Wahid, Pekalongan, Indonesia*
efriati.annisa.hakim25050@mhs.uingusdur.ac.id [1*], dinda.puji.lestari25069@mhs.uingusdur.ac.id[2],
vanisa.amelia25071@mhs.uingusdur.ac.id[3],imam.prayogopujiono@uingusdur.ac.id[4]

Abstract

The rapid growth of small-scale e-commerce platforms in Indonesia has increased the need to select an appropriate database management system (DBMS) that ensures performance, scalability, and data integrity. This study compares the performance, flexibility, and scalability of MySQL and MongoDB through a Systematic Literature Review (SLR) of 20 scientific articles published between 2021 and 2025. The results show that no database is absolutely superior, since the optimal performance of each system depends on the workload it handles. MongoDB performs better in write-intensive operations, simple data reads, schema flexibility, and horizontal scalability, making it suitable for managing dynamic data such as product catalogs, user sessions, and shopping carts. MySQL is more reliable in maintaining transactional integrity through ACID properties, executing complex relational queries efficiently, and ensuring data reliability, making it the preferred choice for critical data such as financial transactions and inventory management. Based on these findings, this study recommends a Polyglot Persistence strategy, a hybrid approach in which MySQL serves as the source of truth for critical transactional data while MongoDB manages dynamic data requiring high performance and flexibility. This strategy offers evidence-based guidance for developers and system architects building efficient, responsive, and consistent small-scale e-commerce platforms.

Keywords: *Database Performance; E-Commerce; MongoDB; MySQL; Polyglot Persistence; Systematic Literature Review*

1. Introduction

Advances in digital technology have brought significant changes to the way businesses operate across various sectors. E-commerce is one of the fields experiencing the highest growth, owing to its ability to reach consumers widely through online platforms. The rising volume of transactions, user interactions, and product searches generates an ever-increasing accumulation of data over time. This condition places the database management system (DBMS) as a crucial element in maintaining application performance, speed, and stability [1].

The choice of database architecture directly affects application efficiency, the system's capacity to accommodate data growth, and the cost-effectiveness of operations. MySQL, as a relational database, has long been a primary choice because of its ability to maintain data consistency and integrity through the ACID principle (Atomicity, Consistency, Isolation, Durability) [2]. Its well-organized table structure makes MySQL reliable for managing transactional data such as orders, payments, and inventory.

The need for systems capable of handling unstructured data more flexibly gave rise to the NoSQL database paradigm. MongoDB has become one of the most widely adopted implementations because it relies on a document model with an adaptive data structure and supports horizontal scalability through a sharding mechanism [3]. These advantages allow MongoDB to efficiently handle large data volumes, intensive write operations, and dynamic schema changes [4].

Small-scale e-commerce platforms are often faced with a dilemma in determining the most suitable database system. MySQL is considered superior in maintaining transaction integrity and the efficient execution of complex queries, while MongoDB is more adaptive for managing product data and user activity that changes frequently [5]. The fundamental differences in the architectural design of both systems affect how various query patterns commonly found in e-commerce environments are executed, such as product catalog management, customer transactions, and data browsing activities.

A number of previous studies have examined the comparison between these two database systems in various testing contexts. Several researchers [7], [8] found that MongoDB has an advantage in write-operation speed, while MySQL shows better performance in analytical queries involving relations between tables. Another study [9] examined the performance comparison of both systems within a REST API architecture widely used in modern e-commerce applications. However, studies that specifically address performance comparison based on query patterns typical of small-scale e-commerce remain very limited and have not been examined in depth.

The core problem addressed in this study lies in the limited comprehensive understanding of the performance differences between MySQL and MongoDB in handling data access patterns commonly used in small-scale e-commerce platforms. Few studies elaborate in detail on how the architectural differences between the two systems affect response time, throughput, and query execution efficiency. Therefore, a

systematic study is needed to provide a clear and measurable picture of the performance of both systems in the operational context of small-scale e-commerce.

This study aims to evaluate the performance of MySQL and MongoDB based on query patterns commonly applied in small-scale e-commerce. The analysis is conducted to understand the characteristics of each system in terms of speed, scalability, and schema flexibility. The results of this study are expected to produce database architecture recommendations that are efficient and aligned with the characteristics of diverse workloads.

Based on this formulation, the study was systematically designed using a Systematic Literature Review (SLR) approach by identifying, analyzing, and synthesizing 20 scientific articles published between 2021 and 2025 that discuss the performance of MySQL and MongoDB. The SLR process was carried out through literature searches in academic databases such as Google Scholar and IEEE Xplore using relevant keywords, followed by a selection stage based on predetermined inclusion and exclusion criteria. Each article that passed the selection process was analyzed based on performance metrics covering response time, throughput, scalability, and schema flexibility. The evaluation results were grouped according to query pattern type to identify consistent patterns and trends. The synthesis of all findings was then used to formulate database architecture recommendations suited to the operational characteristics of small-scale e-commerce, while also supporting the implementation of systems that are efficient and adaptive to the evolving needs of digital business.

The final results of this study are expected to provide a real contribution to developers, researchers, and system designers in determining a well-targeted data management strategy. This study also enriches the literature on the evaluation of database management system performance in the context of small-scale e-commerce, and serves as a foundation for building systems that are responsive, stable, and resource-efficient. Furthermore, this study is also expected to bridge the gap between academic studies and practical needs in the field. For small-scale e-commerce businesses with limited resources, choosing the right database system is not merely a technical decision but also a strategic one that directly affects business continuity and competitiveness. By providing evidence-based, scenario-oriented analysis, this study is expected to serve as a practical reference that helps stakeholders make more accurate, measurable, and sustainable decisions in managing their data infrastructure.

2. Research Methodology

2.1. Research Stages

This study uses the Systematic Literature Review (SLR) approach as its main method, given its strength in identifying, evaluating, and synthesizing research findings transparently and reproducibly [15]. This approach enables a thorough analysis of empirical evidence from various studies without the need for primary experiments, making it highly relevant for assessing technology performance based on the existing literature.

The research implementation refers to an established framework in the field of software engineering and information systems. The initial stage begins with formulating the following research questions:

a. How does the performance of MySQL and MongoDB compare in handling write-operation (create, update) query patterns in an e-commerce environment?

b. How does the performance of both systems compare in read-operation query patterns, both simple and complex?

c. Which architectural factors are most influential in determining the choice between MySQL and MongoDB for small-scale e-commerce platforms?

Literature searches were conducted on Google Scholar and IEEE Xplore using a combination of keywords: 'MySQL vs MongoDB performance', 'SQL vs NoSQL benchmark', 'MongoDB MySQL e-commerce', and 'database performance query pattern'. The selection process was carried out through identification, title-abstract screening, full-text eligibility assessment, and final article determination.

Inclusion criteria were set to ensure that the selected articles were empirical or comparative studies published between 2021 and 2025 and discussed relevant performance metrics, such as response time and throughput. Meanwhile, exclusion criteria were applied to filter out non-scientific material, studies outside the specified time range, and research that did not include a direct comparison between MySQL and MongoDB. Through this selection process, 20 core articles were obtained as the main subject of the study, including significant works from [2], [4], and [7].

Data extraction was carried out in a structured manner by recording the research methodology, the volume of data used, performance metrics, and quantitative findings into a prepared standard form. The data were then synthesized narratively by grouping findings according to query pattern type and the performance metrics measured. Consistent trends and patterns were identified to build a coherent argument, while conflicting findings were examined further to understand the underlying contextual factors, such as differences in system configuration and workload characteristics. The results of this synthesis subsequently formed the main foundation for compiling the results and discussion sections of the study.

To ensure the validity and reliability of the SLR process, every research stage was carefully and consistently documented. Every decision in the article selection, data extraction, and findings synthesis process was explicitly recorded to minimize potential bias and increase research transparency. Validation was conducted by cross-checking findings from various different sources to verify the consistency of results. This approach ensures that the conclusions drawn are not only representative but also methodologically accountable.

Overall, the methodological design applied in this study was developed to produce an in-depth, objective, and evidence-based understanding of the performance comparison between MySQL and MongoDB in the context of small-scale e-commerce. By integrating various findings from diverse literature, this study not only strengthens the theoretical foundation but also provides a practical contribution that can serve as a reference in decision-making related to selecting the database management system most suitable for the needs and operational characteristics of e-commerce platforms in Indonesia.

3. Results and Discussion

3.1. Analysis of Literature Review Results

A systematic review of 20 relevant scientific articles from the 2021-2025 period revealed findings that are both consistent and varied regarding the performance comparison between MySQL and MongoDB, particularly in the context of query patterns commonly found in

small-scale e-commerce applications. The collected data provide a comprehensive picture of the advantages and limitations of each database management system, including the operational conditions under which each system shows its best performance.
Through the literature search and selection process, a total of 20 scientific articles were identified as sources that met all the established inclusion criteria. Each article was critically reviewed to extract information on the comparison focus raised, the methodological approach used by each researcher, and the main findings directly relevant to query patterns in the e-commerce environment. A summary of the analysis results from the entire literature is presented in a structured manner in Table 1, which contains the comparison focus and key findings of each study reviewed.

Table 1: Synthesis of MySQL and MongoDB Literature Review (2021-2025)

Author/Year	Comparison Focus	Key Findings
Setiawan (2025)	REST API performance (response time, throughput)	MySQL outperforms in almost all REST API performance metrics [7].
Halimi, Sudarmanto, Utami, Kusnawi (2021)	Query response time performance	MongoDB excels in write and simple read operations [9].
Zulfahrizan, Tanjung, Al-Kautsar, Kiswanto (2025)	MySQL performance on virtualization	Deployment optimization (virtualization) is crucial for MySQL performance [16].
Ilham, Setiawan (2026)	NoSQL benchmarking	NoSQL (MongoDB) is more flexible and faster for semi-structured data [17].
Arini, Andrew, Baharsyah (2025)	MySQL load balancing	MySQL scalability requires a complex load-balancing architecture [18].
Ramadhan, Senubekti, Suhendri (2025)	SQL vs NoSQL literature study	The choice depends on the need: SQL for integrity, NoSQL for scalability [15].
Pujas, Amar, Arif, Mustamin (2024)	SQL vs NoSQL performance	MongoDB excels at create and read due to replica use and smaller size [12].
Andrianto, Suyatno (2024)	REST API load-testing performance	MongoDB excels in load testing for write and simple read operations [10].
Maysanjaya, Dermawan (2024)	Database management theory	Provides the theoretical basis for ACID vs BASE [11].
Haryanto, Ardiansyah, Kurniasih (2024)	Introduction to NoSQL vs relational	NoSQL overcomes the scalability limitations of RDBMS [6].
Győrödi, Dumşe-Burescu, Zmaranda, Győrödi (2022)	CRUD comparison	MongoDB is faster in CRUD operations as data volume increases [2].
Matallah (2021)	Analytical query comparison	MySQL excels at analytical queries (aggregation/JOIN) [13].
Kastowo, Raharjo (2022)	Data storage comparison	MongoDB is more flexible for unstructured data [19].
Mumtahana (2022)	Transaction design optimization (CRUD)	MongoDB has faster running time for Insert, Update, Delete, Search [5].
Putra, Purwaningrum, Winata (2022)	Query response-time performance	MongoDB is faster for write and simple read operations [20].
Naufal, Nurkhodijah, Anugrah, et al. (2022)	Query response-time performance	MongoDB excels in response time for write and simple read operations [4].
Sudarsono (2021)	Performance comparison	Performance trade-off: MariaDB (SQL) excels at complex reads [21].
Saputra (2021)	Administration module speed	MySQL can provide adequate speed for small-scale applications [22].
Winata, Putra (2021)	Query response-time performance	PostgreSQL (SQL) excels in almost all scenarios [23].
Sinaga, Charibaldi, Cahyana (2023)	CRUD response-time comparison	MongoDB excels in CRUD operations compared to other NoSQL systems [24].

Based on the data presented in Table 1, a fairly clear performance distribution pattern can be observed between the two database management systems compared. MySQL proved superior in handling complex operations, relational query execution, and data integrity maintenance. Conversely, MongoDB showed higher performance in write operations, simple data reads, and flexible schema management. These findings confirm that no single database system can be considered absolutely superior under all conditions. The optimal choice is highly determined by the type of operation performed, the volume of data processed, and the level of consistency required in a particular usage context. In addition, these findings also help fill a research gap from previous studies that had not yet deeply examined the performance of both database systems in the context of query patterns typical of small-scale e-commerce platforms.
Workload characteristics proved to be the most dominant determining factor in the database system selection process for small-scale e-commerce platforms. MySQL is able to maintain low and stable response times for analytical query needs and stock data management. Meanwhile, MongoDB shows a more pronounced advantage in user session management, click-activity logging, and the management of product catalogs that are dynamic and frequently updated.

3.2. Performance Comparison Based on Key Metrics

Referring to the literature analysis results conducted, the next stage is a further synthesis aimed at grouping the performance of MySQL and MongoDB based on the metrics most commonly encountered in the operations of small-scale e-commerce platforms. The synthesis results of the performance comparison of both systems are presented in a structured manner in Table 2, which covers six key metrics as assessment parameters, namely write-operation performance, simple read-operation performance, complex read-operation performance, data integrity and consistency, scalability and throughput, and schema flexibility.

Table 2: MySQL vs MongoDB Performance Comparison on Small-Scale E-Commerce Query Patterns

Performance Metric	MySQL	MongoDB	Implication for Small-Scale E-Commerce
Write Operations	Moderate, schema-validation overhead	Very high, consistent, schema-on-read	Activity log recording, shopping cart management, product reviews. MongoDB excels.
Simple Read Operations	Fast, efficient for structured data	Very fast, single read operation	Displaying product detail pages, user profiles, session data. MongoDB excels.
Complex Read Operations (Aggregation/JOIN)	Very fast, efficient for multi-table queries	Moderate, performance decreases on deep JOINs	Monthly sales reports, business data analysis, customer segmentation. MySQL excels.
Data Integrity & Consistency	High, ACID compliant	Lower, BASE, eventual consistency	Payment processing, stock management, financial transactions. MySQL is critical here.
Scalability & Throughput	Primarily vertical, horizontal is complex	Efficient horizontal scaling via sharding	Handling traffic spikes. MongoDB excels for data growth.
Schema Flexibility	Low, changes are costly	Very high, dynamic	Rapid feature iteration, adding product variants, user preferences. MongoDB excels.

Based on the performance synthesis presented in Table 2, the analysis results reveal that implementing both database management systems in a hybrid manner is the most optimal solution for small-scale e-commerce platforms. MongoDB has proven superior in handling intensive write operations, simple data reads, and schema flexibility that allows new product attributes to be added without causing downtime. On the other hand, MySQL still shows an advantage in maintaining data integrity and consistency, particularly for financial transactions and operations involving many tables at once.

This balance between access speed and data consistency further reinforces the relevance of the Polyglot Persistence strategy, in which MySQL functions as the main transactional data source, or source of truth, while MongoDB is utilized as a dynamic layer for managing frequently changing data that requires high performance for intensive write operations. MongoDB's advantage in simple CRUD operations is further reinforced by several recent studies, including [2], [4], and [8], which confirm increased efficiency as data volume grows.

MongoDB adopts a schema-on-read architecture that minimizes data-validation overhead, making it highly suitable for handling real-time user activity. Conversely, MySQL continues to show better performance in executing analytical queries and complex relational queries, such as generating monthly sales reports or customer segmentation [13].

The overall analysis confirms a clear trade-off between performance and data integrity. MongoDB offers high flexibility and good scalability, while MySQL provides assurance of transaction integrity and data reliability for critical operations. Combining both allows a system to operate efficiently without sacrificing either consistency or response speed.

In the context of CRUD operations, MySQL, with its ACID principle, ensures that every transaction is fully completed or not executed at all, thereby minimizing the risk of data loss or inconsistency. This becomes especially crucial in operations involving payments, stock management, or order processing with many interrelated entities. Study [22] confirms that MySQL remains the primary choice when data integrity cannot be compromised, although its performance tends to decline under intensive write operations with large data volumes.

MongoDB uses the BASE approach (Basically Available, Soft State, Eventual Consistency), which prioritizes availability and high performance at the expense of instant consistency. Study [13] shows that this approach is well suited for applications tolerant of delayed consistency, such as product catalog management, user session management, or user activity history logging. Nevertheless, using MongoDB for critical transactions should ideally be supported by additional strategies, such as centralized logging systems or audit mechanisms, so that eventual consistency does not potentially cause errors in business processes.

Scalability is a major concern, especially during sudden surges in users, such as during a flash sale promotion. MySQL traditionally relies on vertical scalability by increasing server capacity, which remains fairly adequate for small to medium scale but begins to show limitations as data volume continues to grow. Study [18] notes that MySQL can be optimized through the implementation of load balancing and clustering, although the complexity increases as data grows.

Meanwhile, MongoDB leverages horizontal scalability through a sharding mechanism, namely by adding new nodes to accommodate increasing data loads. Study [9] shows that distributing workload through this mechanism is able to maintain high response speed even as the number of transactions continues to grow. This advantage makes MongoDB more adaptive to rapid and fluctuating data growth, particularly for intensive write operations. The combination of MySQL for critical transactions and MongoDB for write operations and simple reads presents a hybrid strategy capable of balancing performance, consistency, and scalability simultaneously.

Schema flexibility is also an important factor to consider. MongoDB allows the addition of new attributes or changes to document structure without causing downtime, thereby supporting rapid and efficient feature iteration and the addition of new products. Study [12] emphasizes that this capability is essential for e-commerce platforms that must adapt to changing market trends, diverse product variations, and continuously evolving user preferences. MySQL, although less flexible in terms of schema changes, remains superior in maintaining data structure integrity, so that every change to a table must be carefully planned to avoid damaging existing data.

The synthesis of all findings shows a clear pattern: MySQL excels in operations that require high integrity, complex queries, and critical transactions, while MongoDB excels in intensive write operations, simple data reads, schema flexibility, and horizontal scalability. Combining both in a hybrid architecture becomes an ideal solution for small-scale e-commerce platforms, while also addressing the gap in previous research that had not yet examined the performance of both systems in an integrative and comprehensive manner.

The selection of a database system should not be absolute but rather adjusted to specific operational needs. The most optimal strategy for small-scale e-commerce is to use MySQL as the primary transactional database that guarantees data consistency and integrity, and MongoDB for managing dynamic data, intensive write operations, and simple reads to support schema flexibility while accelerating responses to user requests. This approach provides practical guidance that developers and researchers can rely on to optimize the performance of small-scale e-commerce platforms without sacrificing either data integrity or flexibility.

3.3. Architecture Recommendations and Practical Implications

The literature evaluation results reveal a significant correlation between the level of query pattern complexity and the performance of each database system. MySQL shows a notable advantage in queries involving many tables, JOIN operations, and data aggregation, while MongoDB excels in handling simple queries, batch writes, and intensive read and write operations. This finding is consistent and recurs across various studies reviewed, including [5], [9], and [17].

Practically, these findings carry quite significant implications for e-commerce platform development. Developers can implement a query-splitting strategy based on complexity level, namely using MySQL to handle critical and analytical queries, and utilizing MongoDB for dynamic and operational queries. This approach has the potential to reduce system bottlenecks, increase overall throughput, while keeping user satisfaction levels optimal. Beyond merely showing performance differences, the results of this study also provide concrete guidance for designing a structured hybrid architecture, addressing the gap in previous research, and providing a strong scientific foundation for decision-making in database system selection for small-scale e-commerce platforms.

Combining MySQL and MongoDB within a single ecosystem allows a platform to optimally leverage the advantages of each, so that the system remains adaptive, maintains data consistency, and sustains high performance, while accommodating data growth that is rapid and dynamic.

As an additional summary, Figure 1 was created to provide a comprehensive overview of the comparison of key aspects between SQL (MySQL) and NoSQL (MongoDB) based on the literature findings from 2021-2025.

Aspek	SQL (MySQL)	NoSQL (MongoDB)
Struktur Data	Menggunakan model relasional dengan skema ketat. Cocok untuk data yang sangat terstruktur dan membutuhkan konsistensi tinggi.	Model non-relasional yang fleksibel (dokumen JSON). Mampu menangani data semi-terstruktur dan tidak terstruktur yang umum pada *e-commerce* modern.
Kinerja Operasi CRUD	Performa *create* dan *read* cenderung menurun ketika volume data mencapai ratusan ribu hingga jutaan entri. Literatur menunjukkan *bottleneck* pada *insert* dan *update* massal.	CRUD jauh lebih cepat dan stabil pada skala besar. Beberapa studi melaporkan peningkatan performa 30–70% pada operasi *write-heavy* dan *read-heavy* untuk dataset besar.
Kinerja Query Relasional	Sangat unggul untuk *query* kompleks berbasis *JOIN*, agregasi multilevel, dan *constraint* relasional yang kuat.	Kurang optimal untuk *query* relasional kompleks. *JOIN* antar koleksi tidak sekuat SQL sehingga membutuhkan denormalisasi atau *pipeline* agregasi khusus.
Skalabilitas	Mendukung *vertical scaling* tetapi kurang efisien dalam *horizontal scaling*. Ketergantungan pada relasi tabel membuat distribusi data lebih sulit.	Mendukung *horizontal scaling* secara *native* (*sharding*). Performa tetap stabil meskipun data tumbuh sangat cepat, sesuai pola *traffic e-commerce*.
Efisiensi Penyimpanan	Umumnya menghasilkan *file* penyimpanan lebih besar karena normalisasi, indeks relasional, dan redundansi struktur tabel.	Lebih efisien dalam penyimpanan. Studi literatur menunjukkan penciutan ukuran 20–40% dibanding SQL untuk dataset *e-commerce* berukuran sama.
Kecepatan Waktu Respon	Waktu respon meningkat tajam pada *query* baca massal dan transaksi paralel berskala besar.	Waktu respon baca dan tulis jauh lebih cepat pada data besar, terutama pada skenario *high concurrency* dan *bulk operations*.
Kompleksitas Implementasi	Struktur yang terstandarisasi membuat implementasi lebih mudah jika data memiliki relasi jelas. Migrasi skema cukup kaku.	Desain skema fleksibel, tetapi bisa menjadi kompleks jika tidak direncanakan matang karena potensi inkonsistensi antar dokumen.
Kasus Penggunaan	Sistem keuangan, ERP, aplikasi transaksi yang menuntut ACID kuat dan integritas relasional.	E-commerce, aplikasi mobile, analitik big data, *logging*, dan skenario *high-traffic* yang membutuhkan performa baca/tulis tinggi.
Kelemahan Utama	Performa menurun pada data besar, kurang efisien untuk kebutuhan skalabilitas besar dan beban kerja *write-heavy*.	Tidak optimal untuk *query* relasional kompleks dan membutuhkan perancangan skema denormalisasi yang lebih hati-hati.

Fig. 1: Comparison of SQL (MySQL) and NoSQL (MongoDB) Aspects

This synthesis shows that MySQL and MongoDB each offer distinct advantages that are highly influenced by the context of implementation and the type of operation performed. Therefore, understanding the performance characteristics of both systems becomes a crucial foundation for designing an efficient, reliable, and flexible data management strategy for the needs of small-scale e-commerce.

4. Conclusion

This study shows that MySQL is a highly suitable choice for operations requiring a high level of data integrity along with complex and structured query handling. On the other hand, MongoDB proves superior in managing operations with high write frequency, simple data reads, flexible data structure management without a fixed schema, and efficient horizontal scaling capability.

Combining both database systems within a single hybrid architecture opens a significant opportunity for small-scale e-commerce platforms to leverage the strengths of each technology simultaneously and complementarily. This integrated approach allows overall data consistency to be maintained, improves overall system performance, and provides a better capability to accommodate dynamic, unpredictable, and continuously growing data volumes as user activity increases.

As a recommendation for future research directions, it is strongly suggested that researchers examine the direct implementation of the Polyglot Persistence strategy in e-commerce environments operating at a much larger and more complex scale. Such a study should cover aspects of optimizing data-replication mechanisms between database systems, a thorough assessment of the operational costs incurred from implementing this architecture, and an in-depth analysis of the extent to which this hybrid approach directly affects the quality of the end-user experience.

Acknowledgement

The authors would like to thank their academic supervisor for the guidance, suggestions, and input provided throughout the preparation of this article. The authors also extend their appreciation to the journal, the editors, and the reviewers for their valuable feedback and review process, which made it possible for this article to be completed and published properly. It is hoped that this article will be beneficial to readers and contribute to the development of knowledge.

References

[1] M. N. Ilham, A. F. Setiawan, I. Prawira, D. Purnomo, D. A. Nugroho, and I. P. Pujiono, "Evaluasi performa MySQL dan MongoDB pada query pattern e-commerce skala kecil berbasis literatur," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 10, no. 1, pp. 15-21, 2026.

[2] C. A. Győrödi, D. V. Dumşe-Burescu, D. R. Zmaranda, and R. Győrödi, "A Comparative Study of MongoDB and Document-Based MySQL for Big Data Application Data Management," *Big Data and Cognitive Computing*, vol. 6, no. 2, p. 49, 2022, doi: 10.3390/bdcc6020049.

[3] A. A. Muhamad, D. A. Kusuma, I. Renaldi, and A. Wibowo, "Studi Literatur Komparasi SQL dan NoSQL dalam Pemilihan Basis Data Ideal untuk Skalabilitas Tinggi," *IKN: Jurnal Informatika dan Kesehatan*, vol. 2, no. 1, pp. 37-48, 2025.

[4] A. P. Nandy Naufal, Siti Nurkhodijah, Gregian Bayu Anugrah, R. F. Muhammad Ihsan Rabbani, Firgi Ahmad Dilla, and Tiara Nurhaliza Anggraeni, "Analisa Perbandingan Kinerja Response Time Query MySQL dan MongoDB," *Jurnal JITEK*, vol. 2, no. 2, pp. 158-166, 2022.

[5] H. A. Mumtahana, "Optimization of Transaction Database Design with MySQL and MongoDB," *Sinkron*, vol. 7, no. 3, pp. 883-890, 2022, doi: 10.33395/sinkron.v7i3.11528.

[6] M. K. Budi Haryanto and Achmad Ardiansyah, "Pengenalan Database NoSQL dan Perbandingannya dengan Database Relasional," *Jurnal IPSIKOM*, vol. 12, no. 1, 2024.

[7] I. Setiawan, "Comparative Study of MySQL and MongoDB Performance on Node.js REST API Using Chinook Database," *Jurnal Media Computer Science*, vol. 4, no. 2, pp. 339-348, 2025, doi: 10.37676/jmcs.v4i2.

[8] H. Susanto, "Analisis Perbandingan Penggunaan MongoDB dengan MySQL pada Sistem Informasi Akademik," Institut Teknologi Adhi Tama Surabaya, 2022.

[9] K. Ahmad Halimi, Ardi Sudarmanto, and Ema Utami, "Analisis Perbandingan Kinerja Waktu Respon MySQL 8.0 dan NoSQL MongoDB Menggunakan RestAPI NodeJS pada Studi Kasus Kelas Online," *Jurnal Informatika*, vol. 10, no. 1, pp. 26-33, 2021.

[10] L. D. Andrianto and D. F. Suyatno, "Analisis Performa Load Testing antara MySQL dan NoSQL MongoDB pada RestAPI NodeJS Menggunakan Postman," *JEISBI (Journal of Emerging Information Systems and Business Intelligent)*, vol. 5, no. 1, pp. 18-26, 2024.

[11] I. M. D. M. K. T. Dermawan, *Manajemen Basis Data: Teori dan Implementasi*. 2024.

[12] M. Sofyan, D. Pujas, M. I. Amar, M. E. Arif, and M. M. Mustamin, "Pengukuran Kinerja Database SQL dan NoSQL pada Aplikasi E-Commerce," *Jurnal Fokus Elektroda: Energi Listrik, Telekomunikasi, Komputer, Elektronika dan Kendali*, vol. 9, no. 1, pp. 12-15, 2024.

[13] A. Gomes et al., "An Empirical Performance Comparison between MySQL and MongoDB on Analytical Queries in the Comex Database," in *Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 23-26, 2021, doi: 10.23919/CISTI52073.2021.9476623.

[14] V. R. Putri, C. P. Wibowo, and H. A. Setia, "Analisis Perbandingan Kecepatan Privileges Index pada Database Oracle dan Database MySQL," in *Prosiding Seminar Nasional Teknologi dan Sistem Informasi (SITASI)*, pp. 447-455, 2023.

[15] C. Ramadhan, M. A. Senubekti, S. Suhendri, and U. T. Digital, "Perbandingan SQL dan NoSQL: Studi Literatur tentang Keunggulan, Kelemahan, dan Penerapannya," *Jupiter: Publikasi Ilmu Keteknikan Industri, Teknik Elektro dan Informatika*, vol. 3, no. 3, pp. 1-6, 2025, doi: 10.61132/jupiter.v3i3.785.

[16] A. Zulfahrizan, M. Raffi Akbar Tanjung, M. Zidane Al-Kautsar, and D. Kiswanto, "Analisis Perbandingan Performa MySQL pada Lingkungan Virtualisasi pada Linux Debian dan Ubuntu Berbasis Web Server Nginx," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 1, pp. 1107-1112, 2025, doi: 10.36040/jati.v9i1.12545.

[17] J. Cordelie, "Benchmarking Basis Data NoSQL pada PT Infracom Technology," Laporan, Universitas Multimedia Nusantara, 2025.

[18] A. Arini, A. Fiade, and R. Baharsyah, "Perbandingan Load Balancing Router MySQL dan HAProxy Menggunakan Sysbench dan Cluster InnoDB pada Sistem Operasi CentOS," *Cyber Security dan Forensik Digital*, vol. 8, no. 1, pp. 17-24, 2025, doi: 10.14421/csecurity.2025.8.1.5004.

[19] D. Kastowo and S. Raharjo, "Analisis Perbandingan Penyimpanan Data Rekam Medis Elektronik Berstandar FHIR pada Sistem Basis Data: BigchainDB, MySQL dan MongoDB," *Jnanaloka*, pp. 37-53, 2023, doi: 10.36802/jnanaloka.2023.v4-no01-37-4x.

[20] R. H. W. Yudha Yunanto Putra and Oktania Purwaningrum, "Perbandingan Performa Respon Waktu Kueri MySQL, PostgreSQL, dan MongoDB," *Jurnal Sistem Informasi dan Bisnis Cerdas (SIBC)*, vol. 15, no. 1, pp. 39-48, 2022.

[21] D. Sudarsono, M. D. Anggraini, C. Andriyani, A. N. Widiyanto, F. Ilmawan, and E. Utami, "Perbandingan Performansi antara MongoDB dan MariaDB," *Jurnal Fahma*, vol. 19, no. 2, pp. 1-11, 2021.

[22] H. Saputra, "Pengembangan Sistem Informasi Akademik Pondok Pesantren Sumber Barokah Karawang untuk Meningkatkan Kecepatan Modul Administrasi Keuangan," Universitas Budi Luhur, 2021.

[23] Y. Y. P. Rivaldo Hadi Winata, "Analisis Perbandingan Performa Waktu Respons Kueri antara MySQL PHP 7.4.2, PostgreSQL, dan MongoDB," in *Prosiding Seminar SITASI*, pp. 38-47, 2021.

[24] T. L. Sinaga, N. Charibaldi, N. H. Cahyana, and F. T. Industri, "Perbandingan Waktu Respon Aplikasi Database NoSQL Elasticsearch dan MongoDB pada Pengujian Operasi CRUD," *JISKA (Jurnal Informatika Sunan Kalijaga)*, vol. 8, no. 1, pp. 22-35, 2023.