WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

A Practical Evaluation of Sql and Nosql Database Systems for Tertiary Institutions in Zamfara State: Security, Cost, and Real-World Deployment Considerations

Mustapha Abubakar[1], Faru Abubakar Abdullahi[2]

[1]Department of Computer Science, Faculty of Science Zamfara State University, Talata Mafara,
[2]Department of computer Science, Faculty of Science and Technology, Federal Polytechnics Kaura Namoda,
*** Email**: mayanchi0@gmail.com

ABSTRACT

This study presents a practical evaluation of SQL and NoSQL database systems in tertiary institutions in Zamfara State, Nigeria, with a focus on security, cost, and real-world deployment considerations. While SQL databases like MySQL and PostgreSQL have been the traditional choice for educational institutions due to their robust data integrity and security features, NoSQL systems such as MongoDB and CouchDB have gained popularity due to their flexibility and scalability, particularly for handling large volumes of semi-structured and unstructured data. Despite the increasing adoption of NoSQL systems, there is limited research on their real-world performance and deployment challenges in resource-constrained environments.

The research follows a mixed-methods approach, combining quantitative benchmarking tests and qualitative field observations. Four database systems—MySQL, PostgreSQL, MongoDB, and CouchDB—were selected for evaluation based on their relevance and usage in academic settings. Performance metrics, security features, cost of ownership, and usability were analyzed in three higher institutions in Zamfara State, using a combination of automated tools and stakeholder interviews. The results reveal important insights regarding the operational strengths and weaknesses of both SQL and NoSQL systems, emphasizing that while NoSQL databases offer superior scalability and flexibility, they often lack mature security features compared to their SQL counterparts.

The study recommends a hybrid approach, where institutions deploy both types of databases depending on specific application needs, such as SQL for administrative and financial systems and NoSQL for e-learning platforms and student activity tracking. Furthermore, it highlights the importance of security training, cost-effective deployment strategies, and continuous performance monitoring in ensuring the successful adoption of database technologies in developing countries. The findings contribute to the broader discourse on database selection and provide actionable guidelines for IT administrators and policymakers in Nigeria and similar resource-limited contexts.

1. INTRODUCTION:

The global shift toward digitalization in higher education has accelerated the need for efficient, scalable, and secure data management systems. Universities and colleges increasingly depend on database technologies to handle critical operations such as admissions, course registration, finance, human resources, and e-learning platforms. Effective data management systems not only improve administrative efficiency but also enhance decision-making, transparency, and student services (Elmasri & Navathe, 2015).

Traditionally, relational database management systems (RDBMS) such as MySQL and PostgreSQL have dominated institutional IT infrastructures due to their strong adherence to ACID (Atomicity, Consistency, Isolation, and Durability) principles and relational schema design. However, with the explosion of semi-structured and unstructured data and the demand for real-time analytics, NoSQL databases such as MongoDB and CouchDB have gained popularity. These systems offer greater flexibility, horizontal scalability, and superior performance for large-scale or non-tabular data applications (Sadalage & Fowler, 2012; Cattell, 2011).

Despite this technological evolution, selecting an appropriate database system for specific institutional needs remains a complex decision. Prior studies have primarily focused on comparing SQL and NoSQL systems based on performance and scalability under experimental workloads (Floratou et al., 2012; Hecht & Jablonski, 2011). However, practical aspects such as deployment cost, security measures, usability, and administrative overhead are often overlooked—especially in resource-constrained educational environments like those in developing countries.

In a previous study by Abubakar (2024), SQL and NoSQL systems were evaluated in selected higher institutions in Zamfara State using performance-based benchmarking. The study revealed important insights about structural and processing efficiencies but did not address essential operational concerns,

including data security, real-world deployment costs, and user adoption challenges. This follow-up study seeks to bridge these gaps by providing a broader evaluation framework that integrates security features, usability experiences, and cost-effectiveness into the decision-making matrix for database adoption.

Moreover, there is a pressing need for localized research that reflects the realities of institutions in Northern Nigeria, where limited technical expertise, infrastructure variability, and funding constraints pose unique challenges. This study aims to support IT administrators and policymakers

in making data-informed choices by comparing SQL and NoSQL databases through real-life deployment case studies in Zamfara State. By doing so, it contributes to both the academic discourse and practical solutions in the field of educational information systems.

2. LITERATURE REVIEW:

Over the past decade, the database management landscape has undergone significant evolution driven by changing data formats, increased demand for real-time analytics, and the need for scalable data processing systems. This transformation has been extensively documented in the literature, with researchers comparing SQL and NoSQL systems from various perspectives including scalability, consistency, query performance, and architecture (Cattell, 2011; Sadalage & Fowler, 2012).

Relational databases are grounded in the relational model and use structured schemas to enforce data integrity. They are widely used in enterprise and academic environments due to their mature ecosystems and support for ACID transactions (Elmasri & Navathe, 2015). However, their rigid schema and vertical scalability have become limitations in the face of modern demands for dynamic and distributed data handling.

In contrast, NoSQL databases are designed to handle large volumes of diverse and rapidly changing data. They offer BASE (Basically Available, Soft State, Eventually Consistent) properties and are categorized into key-value stores (e.g., Redis), document stores (e.g., MongoDB), column-family stores (e.g., Cassandra), and graph databases (e.g., Neo4j). MongoDB and CouchDB have emerged as prominent document-based NoSQL databases, lauded for their flexibility, ease of use, and ability to scale horizontally (Strauch, 2011; Li & Manoharan, 2013).

Multiple comparative studies highlight trade-offs between SQL and NoSQL databases. For instance, Hecht and Jablonski (2011) emphasized that NoSQL databases excel in distributed environments but often lack strong transactional support. Floratou et al. (2012) investigated the performance of various systems under MapReduce workloads and noted that while NoSQL systems offer speed, they require careful configuration to achieve consistency and fault tolerance.

Security is another critical dimension, especially in educational institutions handling sensitive student and staff data. Alsubaiee et al. (2014) explored the security capabilities of emerging NoSQL databases and found many lacked built-in encryption, auditing, and access control mechanisms. Conversely, SQL systems like PostgreSQL are equipped with mature authentication, role-based access control, and integration with security standards (Kumar & Garg, 2017).

Additionally, literature on database selection in the context of developing countries indicates that infrastructure limitations and technical skill gaps significantly influence technology adoption. Ogundele et al. (2020) noted that universities in Nigeria often struggle with integrating advanced IT systems due to limited funding, inadequate training, and lack of localized best practices. As such, any evaluation framework must account for environmental constraints alongside technical performance.

Building on the benchmark work by Abubakar (2024), which focused on SQL and NoSQL performance in controlled settings, this study expands the literature by considering usability, cost-effectiveness, and security—factors that are often overlooked in academic comparisons but are vital for real-world deployments.

3. METHODOLOGY

This study adopts a mixed-methods research design that combines quantitative benchmarking with qualitative field observations and stakeholder interviews to evaluate SQL and NoSQL database systems. The methodology comprises the following key phases:

3.1 Research Framework

The evaluation framework was designed around four critical dimensions:

1. **Performance Efficiency** (query execution time, data write/read speed)
2. **Security Features** (authentication, data encryption, role-based access)
3. **Cost of Ownership** (deployment, licensing, maintenance)
4. **Usability and Administrative Overhead** (ease of setup, learning curve, error handling)

These dimensions align with institutional priorities in higher education settings, especially in resource-constrained environments.

3.2 Selection of Databases

Four popular database systems were selected for analysis based on their relevance and prevalence in academic and commercial use:

- **SQL (Relational):** MySQL, PostgreSQL
- **NoSQL (Non-relational):** MongoDB, CouchDB

3.3 Data Collection Environment

Three higher institutions in Zamfara State—two universities and one polytechnic—were selected for real-world deployment testing. Each institution provided anonymized access to a department-level database system (e.g., student records, course management, and finance).

3.4 Experimental Setup

Each selected database was deployed using Docker containers on identical virtual server configurations:

- **CPU:** Quad-Core, 2.4GHz
- **RAM:** 8GB
- **Storage:** 256GB SSD
- **OS:** Ubuntu 22.04 LTS

The following benchmarking tools were used:

- **Sysbench** – for SQL performance testing
- **YCSB (Yahoo! Cloud Serving Benchmark)** – for NoSQL evaluation
- **OWASP Dependency Check** – for security vulnerability analysis

3.5 Performance Testing Process

We tested each system using a standardized workload with a combination of read-heavy (70% reads, 30% writes) and balanced (50/50) transactions. Results were recorded over five trial sessions and averaged for consistency.

3.6 Stakeholder Interviews

Structured interviews were conducted with IT personnel, database administrators, and system users to gather qualitative insights into:

- Perceived system usability
- Common administrative challenges
- Security awareness and practices
- Training and documentation needs

3.7 Cost Analysis Approach Cost components were calculated using both real institutional expenses and online market rates, including:

- Software licensing (if applicable)
- Setup and migration costs
- Maintenance overhead (human resources)
- Energy and infrastructure-related expenditures

4. Findings, Results, and Discussion

This section presents the results obtained from the practical evaluation of SQL and NoSQL databases across the selected higher institutions in Zamfara State. The analysis is grouped according to the four dimensions outlined in the methodology: **performance**, **security**, **cost**, and **usability**.

4.1 Performance Evaluation

Using Sysbench (for SQL) and YCSB (for NoSQL), we measured latency, throughput, and query response time under read-heavy and balanced workloads. The average results across five trials are shown in Table 1.

Table 1: Average Query Performance (ms)

Database	Read-Heavy (70R/30W)	Balanced (50R/50W)	Throughput (ops/sec)
MySQL	28 ms	33 ms	1500
PostgreSQL	25 ms	30 ms	1600
MongoDB	15 ms	18 ms	3000
CouchDB	19 ms	21 ms	2700

Key Insight:

NoSQL systems, particularly MongoDB, significantly outperformed SQL systems in terms of throughput and latency. However, PostgreSQL exhibited slightly better consistency and data integrity under simultaneous writes.

4.2 Security Assessment

Security was evaluated using OWASP Dependency check and through institutional policy reviews. The findings are summarized in Table 2.

Table 2: Security Features Assessment

Feature	MySQL	PostgreSQL	MongoDB	CouchDB
SSL/TLS Support	Yes	Yes	Yes	Yes
Role-Based Access Control	Yes	Yes	Limited	Limited
Built-in Auditing	Partial	Yes	No	No
Data-at-Rest Encryption	External Tool	Yes	No	No

Key Insight:

PostgreSQL emerged as the most security-complete solution, offering robust access control and encryption. In contrast, MongoDB and CouchDB require significant manual configuration or external tools to meet standard compliance levels.

4.3 Cost Analysis

Cost analysis included software acquisition, deployment, training, and maintenance. Values were collected from both institution records and online averages (in Nigerian Naira ₦).

Table 3: Total Deployment and Maintenance Costs (Annualized)

Database	Licensing	Setup & Migration	Training	Maintenance	Total
MySQL	Free	₦150,000	₦100,000	₦200,000	₦450,000
PostgreSQL	Free	₦180,000	₦120,000	₦220,000	₦520,000
MongoDB	Free	₦100,000	₦80,000	₦160,000	₦340,000
CouchDB	Free	₦110,000	₦90,000	₦150,000	₦350,000

Key Insight:

NoSQL solutions had lower overall costs, making them attractive to institutions with limited budgets. However, the lack of built-in security features might translate into long-term risks and indirect costs.

4.4 Usability and Administrative Feedback

Stakeholder interviews revealed the following:

- **SQL Systems:** Users appreciated their stability and rich documentation. However, rigid schemas made adaptation to changing institutional needs more difficult.

- **NoSQL Systems:** MongoDB was praised for its ease of use and quick deployment. However, frequent crashes due to misconfigurations and poor security documentation were challenges.

Quote from an IT Officer (Polytechnic):

"MongoDB was easy to set up, but we had to consult multiple forums to fix a role-based access issue. PostgreSQL was more secure, but harder to learn."

4.5 Discussion

The findings reveal a nuanced landscape:

- **Performance:** NoSQL databases (especially MongoDB) offer superior speed and scalability.
- **Security:** PostgreSQL leads with enterprise-grade security, while NoSQL lags without additional setup.
- **Cost:** NoSQL is cheaper in the short term but may pose long-term risks without proper controls.
- **Usability:** MongoDB has a lower entry barrier but requires a steeper learning curve for secure deployment.

Thus, the choice between SQL and NoSQL should be guided not just by performance metrics, but by **institution-specific factors** such as data sensitivity, funding, and administrative skill levels.

4.4 Usability and Administrative Feedback

Stakeholder interviews revealed the following:

- **SQL Systems:** Users appreciated their stability and rich documentation. However, rigid schemas made adaptation to changing institutional needs more difficult.
- **NoSQL Systems:** MongoDB was praised for its ease of use and quick deployment. However, frequent crashes due to misconfigurations and poor security documentation were challenges.

Quote from an IT Officer (Polytechnic):

"MongoDB was easy to set up, but we had to consult multiple forums to fix a role-based access issue. PostgreSQL was more secure, but harder to learn."

4.5 Discussion

The findings reveal a nuanced landscape:

- **Performance:** NoSQL databases (especially MongoDB) offer superior speed and scalability.
- **Security:** PostgreSQL leads with enterprise-grade security, while NoSQL lags without additional setup.
- **Cost:** NoSQL is cheaper in the short term but may pose long-term risks without proper controls.
- **Usability:** MongoDB has a lower entry barrier but requires a steeper learning curve for secure deployment.

Thus, the choice between SQL and NoSQL should be guided not just by performance metrics, but by **institution-specific factors** such as data sensitivity, funding, and administrative skill levels.

5. Recommendations

Based on the evaluation results across performance, security, cost, and usability dimensions, the following recommendations are proposed for tertiary institutions in Zamfara State and similar environments:

1. **Adopt a Hybrid Database Strategy:**

   Institutions should consider deploying both SQL and NoSQL systems based on specific departmental needs. SQL databases like PostgreSQL are ideal for financial and administrative systems requiring data integrity and robust security, while NoSQL systems like MongoDB are well-suited for dynamic applications such as e-learning platforms and student activity tracking.

2. **Invest in Security Training and Tools:**

   Since NoSQL systems generally lack built-in security features, institutions opting for these platforms must allocate resources for security tools and personnel training. Administrators should be trained in best practices including role-based access, data encryption, and audit logging.

3. **Conduct Regular Performance and Cost Audits:**

4. IT departments should establish KPIs to monitor system performance, user satisfaction, and maintenance costs. This will enable informed decisions about whether to scale existing systems or transition to new ones.
5. **Leverage Open-Source and Community Resources:**

   To reduce cost and technical dependency, institutions should utilize open-source tools and participate in developer communities. PostgreSQL and MongoDB have large user bases and extensive documentation that can be leveraged for support.
6. **Implement Pilot Programs Before Full Deployment:**

   Before large-scale implementation, institutions should run pilot programs to understand deployment challenges, adapt configurations, and fine-tune performance under localized conditions.

6. Further Research

While this study has provided a practical and holistic comparison of SQL and NoSQL database systems, several limitations point to future research opportunities:

1. **Scalability Under Real-Time Load:**

   This study evaluated systems under standard workloads. Future research could explore performance under real-time high-concurrency loads such as exam result releases or mass student registration.
2. **Energy Efficiency and Environmental Impact:**

   Considering rising energy costs, future studies could investigate the power consumption of different database configurations, especially for off-grid or solar-powered campuses.
3. **Security Vulnerability Lifecycle Management:**

   A longitudinal study on how quickly SQL and NoSQL systems respond to known vulnerabilities in institutional environments could provide deeper insights into long-term risk management.
4. **User Experience and Human-Centered Design:**

   Further work could evaluate how database architecture affects the end-user experience, particularly academic staff who interact with portals and dashboards built on these systems.
5. **Comparative Studies Across Regions:**

   Expanding this research to include institutions from other parts of Nigeria or sub-Saharan Africa could improve generalizability and identify regional best practices for educational IT infrastructure.

7 CONCLUSSION:

This study has presented a practical, real-world evaluation of SQL and NoSQL database systems within tertiary institutions in Zamfara State. By incorporating performance metrics, security assessments, cost analyses, and user feedback, the research provides actionable insights for decision-makers navigating the complex landscape of database technology selection. While both SQL and NoSQL systems have strengths and weaknesses, a balanced, hybrid approach tailored to institutional needs proves most effective. The findings underscore the importance of contextualized research and localized deployment strategies in advancing educational technology infrastructure.

8. ACKNOWLEDGEMENT

The author acknowledges the support of Zamfara State University and the management of participating institutions for granting access to their IT departments. Special thanks also go to the faculty mentors, database administrators, and all stakeholders who provided valuable insights and feedback during the course of this study. Lastly, I thanks to TETFUND under Institution Based Research (IBR) for supporting and sponsored of this research.

Here are 10 updated True or False questions based on revised document titled *"A Practical Evaluation of SQL and NoSQL Database Systems for Tertiary Institutions in Zamfara State: Security, Cost, and Real-World Deployment Considerations"*:

True or False Questions

1. **True or False:** Traditional relational databases are designed to handle semi-structured and unstructured data more efficiently than NoSQL databases.
2. **True or False:** The study used only quantitative benchmarking methods to evaluate SQL and NoSQL databases.

3. **True or False:** MongoDB and CouchDB are examples of document-based NoSQL databases.
4. **True or False:** One limitation of NoSQL databases highlighted in the study is their generally weaker built-in security features compared to SQL databases.
5. **True or False:** The study was conducted in three higher institutions located in Southern Nigeria.
6. **True or False:** Sysbench was used to test the performance of SQL databases in the study.
7. **True or False:** PostgreSQL supports role-based access control and encryption, making it suitable for sensitive educational data.
8. **True or False:** The evaluation framework in the study excluded cost and usability considerations.
9. **True or False:** The performance tests were conducted using Docker containers on servers with identical hardware configurations.
10. **True or False:** Stakeholder interviews were conducted to understand the administrative and usability challenges of deploying database systems.

Here is the **answer key** for the 10 True or False questions based on your updated document:

✅ **Answer Key**

1. **False** – Traditional relational databases (SQL) are not designed to efficiently handle semi-structured and unstructured data; NoSQL databases are better suited for that.
2. **False** – The study used both **quantitative benchmarking** and **qualitative methods** (e.g., stakeholder interviews).
3. **True** – MongoDB and CouchDB are both **document-based NoSQL databases**, as correctly mentioned in the Literature Review.
4. **True** – The study notes that **NoSQL systems often have weaker built-in security** features compared to SQL databases.
5. **False** – The study was conducted in **Zamfara State, Northern Nigeria**, not in Southern Nigeria.
6. **True** – **Sysbench** was used for performance testing of SQL databases.
7. **True** – PostgreSQL supports **role-based access control and encryption**, making it suitable for secure environments.
8. **False** – The evaluation **included** cost and usability as key factors in the framework.
9. **True** – All database systems were deployed on **identical virtual servers using Docker containers**.
10. **True** – **Stakeholder interviews** were conducted to gather insights into usability, administrative challenges, and security awareness.

REFERENCES

1. Alsubaiee, S., Behm, A., Borkar, V. R., Carey, M. J., Grover, R., Onose, N., & Vernica, R. (2014). AsterixDB: A scalable, open source BDMS. *Proceedings of the VLDB Endowment, 7*(14), 1905-1916. https://doi.org/10.14778/2733004.2733006
2. Cattell, R. (2011). Scalable SQL and NoSQL data stores. *ACM SIGMOD Record, 39*(4), 12–27. https://doi.org/10.1145/2076331.2076333
3. Elmasri, R., & Navathe, S. B. (2015). *Fundamentals of database systems* (7th Ed.). Pearson.
4. Floratou, A., Teletia, N., DeWitt, D. J., Patel, J. M., & Zhang, D. (2012). Can the elephants handle the NoSQL onslaught? *Proceedings of the VLDB Endowment, 5*(12), 1712–1723. https://doi.org/10.14778/2357159.2357165
5. Hecht, R., & Jablonski, S. (2011). NoSQL evaluation: A use case-oriented survey. In *Proceedings of the International Conference on Cloud and Service Computing* (pp. 336–341). IEEE. https://doi.org/10.1109/CSC.2011.11
6. Kumar, S., & Garg, K. (2017). A comprehensive study on security in relational and NoSQL databases. *International Journal of Computer Applications, 166*(5), 9–15. https://doi.org/10.5120/ijca2017913569
7. Li, F., & Manoharan, S. (2013). A performance comparison of SQL and NoSQL databases. In *Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing* (pp. 15–19). IEEE. https://doi.org/10.1109/PACRIM.2013.6650010
8. Ogundele, L. O., Ayediran, A. A., & Jimoh, R. G. (2020). Challenges and prospects of information systems adoption in Nigerian universities. *Journal of Emerging Trends in Computing and Information Sciences, 11*(2), 45–52.
9. Sadalage, P. J., & Fowler, M. (2012). *NoSQL distilled: A brief guide to the emerging world of polyglot persistence*. Addison-Wesley.
10. Strauch, C. (2011). NoSQL databases. *Stuttgart Media University, 53*(10), 1–14.
11. Abubakar, M. (2024). Analyzing and designing an evaluation benchmark for SQL and NoSQL database systems for some selected higher institutions in Zamfara State. *Journal of Applied Computing and Emerging Technologies, 6*(1), 102–112.