

# BENCHMARKING MYSQL AND NOSQL DATABASE ON EGOVBENCH APPLICATION

Nur Aini Rakhmawati, Muhammad Zuhri, Radityo Prasetyanto Wibowo, Anwar Romadhon, Herdy Ardiansyah, Olive Khoirul

*Departement of Information Systems, Institut Teknologi Sepuluh Nopember  
Jalan Raya ITS, Keputih, Sukolilo, Kota Surabaya, Indonesianur.aini@is.its.ac.id, izzur.zuhri@gmail.com,  
radityo.p.w@is.its.ac.id, anwar\_28romadhon@yahoo.co.id, herdy5758@gmail.com,  
olive.klma@gmail.com*

**Abstract-- Egovbench application monitors websites or social media of Indonesia local governments daily. The process of crawling done by Egovbench produces extensive data, which reduced performance in the data processing. Therefore, there is a need for a database solution that has the best performance such as high processing speed and small database size. This study examined the comparison between relational databases and non-relational databases based on selected metrics to obtain the most suitable database solution for Egovbench. The results show that the MySQL database has the advantage of complex query processing and the use of the database with the smallest storage space. MongoDB database has the advantage of low data transfer volumes. Couchbase database has the advantage of short and straightforward query processing with a high number. The evaluation results show that MySQL is more suitable for Egovbench needs, which is the best response time and query per second. MySQL outperformed the other two databases on backup and storage file sizes testing.**

**Keywords:** benchmark, database, crawling, E-government

## I. INTRODUCTION

In recent decades, there has been a revolution in computing and communication, with the indications that technological advances and the use of information technology[1]. The internet is the place for the development and use of information resources that include a sea of global information [2]. For billions of people around the world, the internet has become an essential component of their daily social and business life [3]. Based on a survey conducted by the Indonesian Internet Service Providers Association (APJII), in 2016 there were 132.7 million Indonesians who had used the internet. With a total population of 256.2 million people, more than 52% of Indonesia's population is connected to the internet [4]. With the breadth of the field of information technology, the world moves from the industrial age to the information age. The manifestation of this transformation and the emergence of transitions of terms and concepts have become part of everyday life. The economic field emerged ideas such as E-commerce, E-business, and electronic money, while in the area of public sector appeared the concept of E-government [5]. E-Government is defined as a way for

governments to use the most innovative information and communication technology, especially web-based internet applications, that provides citizens and businesses with easier access to information and government services, to improve service quality and to provide more significant opportunities for participating in democratic institutions and processes [6].

E-government services can be provided through several media, such as websites and social media. Based on the various E-government media, there are differences in quality so that the assessment of different data models is needed to provide a level of consistency, availability, and efficiency in measuring this E-government service using the Egovbench application. Egovbench (E-government benchmarking) is an application that is used to assess, compare and rank websites and social media owned by local governments[7]. Egovbench assessed whether the website and social media were indeed used as E-Government media that served the community or not. Egovbench is used to determine the quality, performance, and quality of official government social media sites and media.

Egovbench crawler obtains information from existing websites or social media. This crawling generates very large data. As a result, it might decrease performance in the data processing. The development of database technology in the last few years has become popular with the presence of non-relational or NoSQL database technology. The NoSQL database appears as a new data management system for managing and processing large volumes of data in an efficient manner [8]. Many companies integrate both databases; users interact with Relational Database (RDB) to handle small scale data, NoSQL databases function as a back-end system for data analysis. NoSQL databases process large volumes of data and produce results in real time such as the analysis of millions of tweets [9]. Unlike the database in general, NoSQL follows different data models and provides a level of consistency, availability, and efficiency.

This research is intended to find out the database performance as the Egovbench storage. We compare MySQL databases, MongoDB, and Couchbase by using four metrics: the response time metrics, query throughput, storage size, and volume of data transfers. Data used for benchmarking is data from social media crawling and government websites. By comparing existing databases, it can be seen how to store

Egovbench data using those three databases. In addition, it can determine the relevance of performance metrics in data processing and can compare performance in Egovbench data processing. By measuring the performance of the database used and conducting website and social ranking of E-government media, it can find out which performance of E-government services are good and which E-government services still need to be improved to provide the best service to the public through the internet.

#### A. Egovbench

Based on the 2014 United Nations E-Government Survey, the definition of E-Government or Electronic Government is the use of information technology and its application by the government to provide information and public services to the public [11]. The Egovbench system currently stored data in the form of documents from content that has been crawled by Egovbench. Data storage is divided into two parts, namely social media data and government website data. Egovbench is an application that is useful for assessing, comparing and ranking e-government websites and government social media. The assessment criteria are:

1. The website must follow the President Instruction Number 3 the Year 2003
2. Website availability
3. Website Update Frequency
4. Social Media availability (Facebook, Twitter, Youtube)
5. Social Media Updates Frequency

#### B. NoSQL Database

NoSQL or Not only SQL or Non-SQL Databases are terms used for formless database types. The NoSQL database follows different data models and provides a level of consistency, availability, and efficiency [9]. Although there are many NoSQL databases on the market, various industry trends show that the top three used today are MongoDB, Apache Cassandra, and HBase [13]. The characteristics below distinguish NoSQL from RDB or Relational Database [14]. NoSQL generally has the following features:

6. Schema-free
7. Easy replication support
8. Simple API
9. Consistent / BASE

MongoDB is a schema-free document-based database written in C++ and developed open-source. The primary purpose of MongoDB is to close the gap between fast and highly scalable key-/value-stores and feature-rich traditional RDBMS. MongoDB has better scalability, meaning MongoDB can connect and utilize multiple devices while managing data [12]. MongoDB databases can provide faster results because MongoDB supports nested documents. By querying the document ID, all documents can be given. In contrary to the relational database, to obtain all the contents of the database a query is needed that involves several tables because they are interrelated.

Couchbase or Membase is a schema-less and document-oriented NoSQL database system and supports zero document

storage in JSON format. Couchbase is intended to run in memory and use as many database nodes as possible to store datasets in RAM before they are written to disk[17].

#### C. Database benchmarking

The database system contributes enormously to the proper and efficient organizational functions and business information [15]. Therefore, choosing the right database with the right features is often a critical decision. Benchmarking is called the process of evaluating a system for the performance of the system. Benchmark is a process to measure performance using specific indicators that produce a value that will be used as a comparison with other objects[16]. The benchmark performance of MongoDB is slightly under Couchbase. The latency of MongoDB increases when additional database servers are performed[13].

Yunhua Gu et al. [12] describes how the type of NoSQL database, namely MongoDB acts as a data storage media in the web crawling process. The purpose of the study was to determine the performance and characteristics of the NoSQL database, especially the MongoDB database. The results obtained from the study show that the MongoDB database does not have the same schemes found in the relational database. In the MongoDB database, there is also no need to specify the number of columns and rows to store data and the absence of a foreign key. Therefore, MongoDB has a more flexible structure. For queries performed can provide faster results because it supports layered or nested documents. By querying the document ID, all the contents of the document can be returned, whereas in the relational database to obtain all the contents of the database a query that involves several tables is needed. On a very large amount of data (above 50 GB), the difference in MongoDB access speed can be up to 10 times faster than the relational database. Also, MongoDB has better scalability, it means that MongoDB can connect and utilize multiple devices while storing and managing data. Yunhua Gu's research has a relationship with this research because it discusses the performance and characteristics of MongoDB in the web crawling process.

Datastax Corporation [13] did a benchmarking the Amazon Web Services EC2 platform by testing using the Cassandra, Couchbase, HBase, and MongoDB databases. The results obtained were that almost all MongoDB performance benchmarks were slightly under Couchbase and latency in MongoDB increased sharply when adding servers to the database.

Joao Ricardo Lourenco, et al. [18] conducted a quality comparison of some of the popular NoSQL databases, Aerospike, Cassandra, Couchbase, CouchDB, HBase, MongoDB, using quality attributes that are measurement parameters, namely Availability, Consistency, Durability, Maintainability, Read & Write Performance, Recovery Time, Reliability, Robustness, Scalability, and Stabilization Time. With the aim to determine the performance of the NoSQL database when measured using several predetermined parameters. The results obtained are that Couchbase is superior in availability, scalability, stabilization time and write-performance when compared with MongoDB which excels in consistency, recovery time, and reliability. The study of Joao

Ricardo Lourenco has a relationship with this research because it discusses the measurement of performance of Couchbase and MongoDB. Comparison between NoSQL Databases can be seen in Table 1.

Paper	AUTHORS	Database
Benchmarking Top NoSQL Databases	End Point Corporation (2013)	<input type="checkbox"/> Cassandra, <input type="checkbox"/> Couchbase, <input type="checkbox"/> HBase, <input type="checkbox"/> MongoDB
Choosing the right NoSQL database for the job: a quality attribute evaluation	Joao Ricardo Lourenc,o, Bruno Cabral, Paulo Carreiro, Marco ~ Vieira, and Jorge Bernardino (2015)	<input type="checkbox"/> Aerospike <input type="checkbox"/> Cassandra <input type="checkbox"/> Couchbase <input type="checkbox"/> CouchDB <input type="checkbox"/> HBase <input type="checkbox"/> MongoDB <input type="checkbox"/> Voldemort
Application of NoSQL Database in Web Crawling	GU Yunhua, SHEN Shu, ZHENG Guansheng (2011)	<input type="checkbox"/> MongoDB

## II. METHOD

### A. Benchmark Metrics

The performance metrics used in this benchmarking are explained as follows:

- Storage size (bytes): to find out how much storage space is needed for each database (DBMS) for the same data type.
- Query per second (Query/second): to determine the level of query throughput that can be served by the database in units of time. We run a set of queries repeatedly and measure the average difference when running a single query.
- Response time (second): to find out the average time needed to run a single query.
- Volume data transfer (bytes): to determine the bandwidth capacity that can be transferred to the related database.

### B. Benchmark Database Configuration

There are several tables that are used on the Egovbench service, i.e. result\_sosmed, content, local government (pemda), and list of website URLs (Daftar link). The database schema can be seen in Figure 1.

The three databases tested have several different characteristics of how to store data because there are differences in structure and features. One of the differences can be seen in Table 2.

TABLE 2. DIFFERENCES IN TERMINOLOGY BETWEEN DATABASES

Terminologi	MySQL	MongoDB	Couchbase
Database	Database	Database	Bucket
Table	Table	Collection	Bucket
Column	Column	Field	Field
Row	Row	Document	Document

In general, the data used consists of 4 parts, namely:

- The social media result table has 26 data fields such as date, regional ID, FB day count, FB content count, etc. This data is used to store the results of local government social media assessments conducted on

the Egovbench application

- Pemda table has seven data fields such as URL, type, FB id, etc. This table is used to keep a list of local government names, local government web addresses, local government types, and social media Facebook Facebook,
- The content table has 4 data fields, namely: regional government id, criteria, documents, and URL. This table is used to store crawling data from local government web content based on predetermined criteria

The Daftar link table has two data fields, namely: local government id and URL link. This section is used to store a list of URLs that have been successfully crawled from the local government web

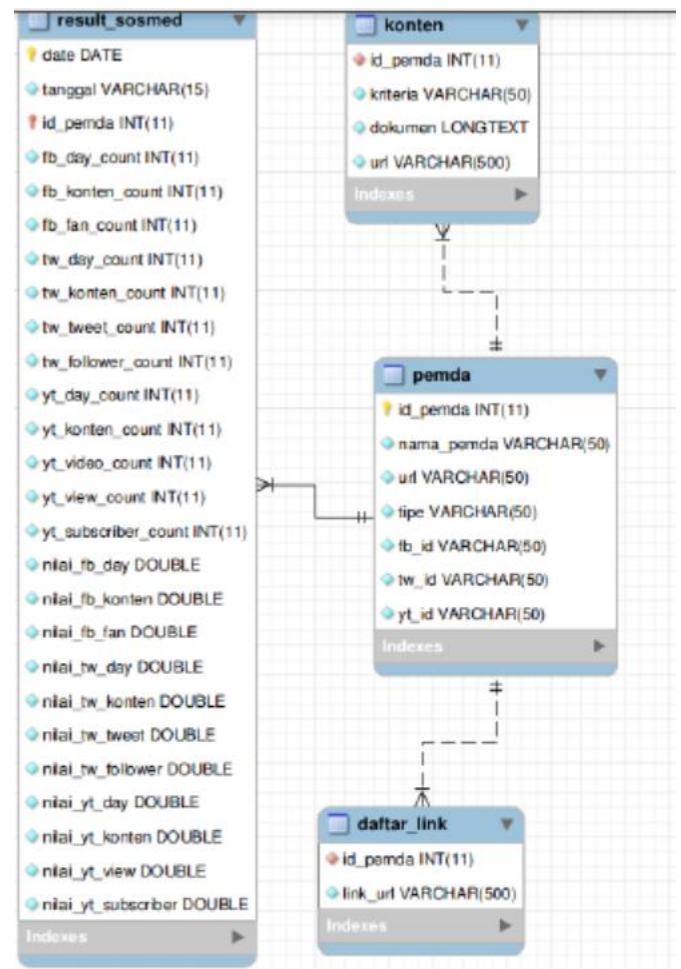


Fig. 1. Database Schema

### C. Set of Evaluation Queries

The queries used in the benchmark are divided into two operations:

- Query Select data from the Egovbench database that has been filled
- Query Insert / Restore data from the database backup

Five queries were selected from the Egovbench application.

The selected queries are queries that often appear in the application. Moreover, we choose simple queries and complex queries. The comparison of the five queries can be seen in Table 3.

TABLE 3. COMPARISON OF THE FIVE QUERIES

Query	Tabel	Jumlah Kolom	Keywords	Operator
1	pemda	2	-	= (equal to)
2	result sosmed	2	ROUND, AS, ORDER BY, LIMIT	*(multiplication) + (addition) = (equal to)
3	result sosmed	2	ORDER BY	>(greater than) = (equal to) AND
4	konten	4	-	= (equal to) AND
5	daftar link	2	ORDER BY	= (equal to)

MySQL queries:

1. SELECT id\_pemda, url FROM 'pemda' WHERE id\_pemda=275
2. SELECT id\_pemda, ROUND( (((nilai\_fb\_day \*52.7794746487477)+(nilai\_fb\_konten \*23.7018937080024)+(nilai\_fb\_fan\*23.5186316432498) )\*0.35284139100933) +(((nilai\_tw\_day \*36.5019011406844)+(nilai\_tw\_konten \*16.3920574566962)+(nilai\_tw\_tweet \*30.8407266582172)+(nilai\_tw\_follower \*16.2653147444022))\*0.33587786259542) +((( nilai\_yt\_day\*42.8571428571429)+(nilai\_yt\_konten \*19.2460317460317)+(nilai\_yt\_view \*18.7996031746032)+(nilai\_yt\_subscriber \*19.0972222222222))\*0.31128074639525)),2) AS totalscore FROM 'result\_sosmed' WHERE date = '2017-06-14' ORDER BY totalscore DESC LIMIT 10
3. SELECT id\_pemda, tw\_tweet\_count FROM 'result\_sosmed' WHERE tw\_tweet\_count>0 AND date='2017-06-14' ORDER BY tw\_tweet\_count DESC
4. SELECT id\_pemda, kriteria, dokumen, url FROM 'konten' WHERE id\_pemda=170 AND kriteria='sejarah'
5. SELECT id\_pemda, link\_url FROM 'daftar\_link' WHERE id\_pemda=162 ORDER BY link\_url ASC

MongoDB Queries:

1. db.pemda.find({ "id\_pemda": 275 }, { "id\_pemda": 1, "url": 1 })
2. db.result\_sosmed.aggregate([ { \$match: { "date": "2017-06-14" } }, { \$project: { "id\_pemda": 1, " totalscore": { \$add: [ { \$multiply: [ { \$add: [ { \$multiply: [ "\$nilai\_fb\_day", 52.7794746487477 ] }, { \$multiply: [ "\$nilai\_fb\_konten", 23.7018937080024 ] }, { \$multiply: [ "\$nilai\_fb\_fan ", 23.5186316432498 ] } ] }, 0.35284139100933 ] }, { \$multiply: [ { \$add: [ { \$multiply: [ "\$nilai\_tw\_day ", 36.5019011406844 ] }, { \$multiply: [ "\$nilai\_tw\_konten", 16.3920574566962 ] }, { \$multiply: [ "\$nilai\_tw\_tweet", 30.8407266582172 ] }, { \$multiply: [ "\$nilai\_tw\_follower", 16.2653147444022 ] } ] }, 0.33587786259542 ] }, { \$multiply: [ { \$add: [ { \$multiply: [ "\$nilai\_yt\_day ", 42.8571428571429 ] }, { \$multiply: [ "\$nilai\_yt\_konten", 19.2460317460317 ] }, { \$multiply: [ "\$nilai\_yt\_view", 18.7996031746032 ] }, { \$multiply: [ "\$nilai\_yt\_subscriber", 19.0972222222222 ] } ] }, 0.31128074639525 ] }, ] } ], { \$sort: { "totalscore": -1 } }, { \$limit: 10 }, { \$project: { "id\_pemda": 1, "totalscore": {

- \$subtract: [ "\$totalscore", { \$divide: [ { \$mod: [ { \$multiply: [ "\$totalscore", 100 ] }, 1 ] }, 100 ] } ] } ] }
3. db.result\_sosmed.find( { "\$and": [ { "tw\_tweet\_count": { "\$gt": 0 } }, { "date": "2017-06-14" } ], { "\_id": 0, "id\_pemda": 1, "tw\_tweet\_count": 1 } ). sort( { "tw\_tweet\_count": -1 } )
4. db.konten.find( { "\$and": [ { "id\_pemda": 170 }, { " kriteria": "sejarah" } ] }, { "\_id": 0, "id\_pemda": 1, "kriteria": 1, "dokumen": 1, "url": 1 } )
5. db.daftar\_link.find( { "id\_pemda": 162 }, { "\_id": 0, "id\_pemda": 1, "link\_url": 1 } ).sort( { "link\_url": 1 } )

Couchbase Query

1. SELECT id\_pemda, url FROM 'pemda' WHERE id\_pemda=275
2. SELECT id\_pemda, ROUND( (((nilai\_fb\_day \*52.7794746487477)+(nilai\_fb\_konten \*23.7018937080024)+(nilai\_fb\_fan\*23.5186316432498) )\*0.35284139100933) +(((nilai\_tw\_day \*36.5019011406844)+(nilai\_tw\_konten \*16.3920574566962)+(nilai\_tw\_tweet \*30.8407266582172)+(nilai\_tw\_follower \*16.2653147444022))\*0.33587786259542) +((( nilai\_yt\_day\*42.8571428571429)+(nilai\_yt\_konten \*19.2460317460317)+(nilai\_yt\_view \*18.7996031746032)+(nilai\_yt\_subscriber \*19.0972222222222))\*0.31128074639525)),2) AS totalscore FROM 'result\_sosmed' WHERE date='Jun 14, 2017' ORDER BY totalscore DESC LIMIT 10
3. SELECT id\_pemda, tw\_tweet\_count FROM 'result\_sosmed' WHERE tw\_tweet\_count>0 AND date='Jun 14, 2017' ORDER BY tw\_tweet\_count DESC
4. SELECT id\_pemda, kriteria, dokumen, url FROM 'konten' WHERE id\_pemda=170 AND kriteria='sejarah'
5. SELECT id\_pemda, link\_url FROM 'daftar\_link' WHERE id\_pemda=162 ORDER BY link\_url

D. Evaluation Scenario

The evaluation steps are as follows:

1. Backing up the database into a file
2. Measuring the size of the backup restore database file from a file calculation of storage used in the database
3. Capturing data packets to calculate the volume of data used by query transactions
4. Running the five queries that have been determined each - 100 times

For each query calculated the time needed to run the query 100 times, the average time needed to run 1 query, and the average query per second.

III. RESULTS AND DISCUSSION

The results of testing the MySQL database, MongoDB, Couchbase are divided into four metrics, including response time, throughput, backup and storage, volume.

The smaller the value of response time, the better the performance because a low response time means that the database can respond to queries given faster. Table 4 shows a comparison of the average response time queries (milliseconds). Couchbase outperforms the other two databases in query 1 and 5 but slower in query 2 and 3 compared to other databases. Couchbase is superior because the nature of the Couchbase database is more memory-centric and more memory

additions greatly help Couchbase's performance, queries 1 and 5 have more simple characteristics than queries 2, 3 and 4. MySQL and MongoDB have almost consistent results in every query. MySQL outperforms MongoDB in each query.

TABLE 4. COMPARISON OF THE AVERAGE RESPONSE TIME QUERY (MILLISECONDS)

Query	MySQL	MongoDB	Couchbase
1	23.99	57.10	22.73
2	27.27	73.93	896.57
3	24.86	74.06	910.41
4	26.17	73.00	32.00
5	24.67	58.10	20.12

The higher the value of query throughput, the better performance because a massive throughput means the database can handle many queries in one second. Table 5 shows a comparison of the average query value per second. Again, Couchbase outperforms the other two databases in query 1 and 5.

TABLE 5. COMPARISON OF AVERAGE QUERY VALUES PER SECOND.

Query	MySQL	MongoDB	Couchbase
1	41.68	17.51	43.99
2	36.67	13.53	1.16
3	40.23	13.50	1.10
4	38.21	13.70	31.25
5	40.53	17.21	49.69

The smaller the value of file size, the better the performance because the small value means that the database can store more data in the same size of storage. Table 6 shows a comparison of the size of the backup and storage files (MB). MySQL outperformed the other two databases in terms of backup files and total database storage. MySQL storage can be smaller because MySQL only converts all data into tabular text and in the query form and stores it in one .sql file whose size is smaller. MongoDB and Couchbase databases store in JSON or BSON (Binary JSON). Moreover, they store for each collection or bucket in different files and folders including metadata information in a separate file. Therefore, the total size of the used file is higher than MySQL.

TABLE 6. COMPARISON OF BACKUP AND STORAGE FILE SIZE (MB).

	MySQL	MongoDB	Couchbase
Backup result file	44.77	186.57	96.84
Database storage size	89.75	310.24	167.48

Table 7 shows a comparison of the volume of TCP data (MB). MongoDB has a slightly better value than MySQL because the output queries of MongoDB are in JSON forms that are shorter than tables generated by MySQL. The Couchbase database uses the most significant volume of data compared to the other two databases because the output generated by Couchbase is JSON which is longer than MongoDB. Besides, the output of the Couchbase contains more metadata.

TABLE 7. TCP DATA VOLUME COMPARISON (MB)

	MySQL	MongoDB	Couchbase
Volume data	4.39	4.27	6.12

#### IV. CONCLUSION

Based on the results of tests conducted on MySQL databases, MongoDB and Couchbase which are divided into four metrics, including response time, throughput, backup and

storage, volume, can be summarized in Table 8. The more star (\*) the more preferred database.

TABLE 8. BENCHMARK RESULTS OF FOUR MYSQL, MONGODB, AND COUCHBASE DATABASE PERFORMANCE METRICS ON THE EGOVBENCH WEB

Database	Response Time	Throughput	Storage Size	Volume Data
MySQL	***	***	***	**
MongoDB	*	*	*	***
Couchbase	**	**	**	*

MySQL database has advantages in query processing. MySQL provides fast response time, the highest query per second, and the smallest storage. MongoDB has advantages in the size of the volume of data queries because it can use fewer network resources in processing queries. Couchbase integrates MySQL and MongoDB databases in terms of measuring response time and throughput metrics in queries 1 and 5. Based on the results of tests that have been carried out, the recommended database for Egovbench is MySQL because it has the best response time and query per second among other databases which is more suitable for Egovbench's service needs.

This research has been attempted and carried out by scientific procedures, but still has flaws, namely:

- This study consists of four metrics, namely response time, throughput, backup and storage, volume. Therefore, it is necessary to consider other metrics and more varied configurations such as indexes on non-relational databases.
- This research is only done on queries 1 to query 5

#### V. ACKNOWLEDGMENT

We would like to thank the Departement of Information Systems, Institut Teknologi Sepuluh Nopember who have helped carry out this research in terms of funding

#### VI. REFERENCES

- [1] B. Shaqiri, "Impact of Information Technology and Internet in Businesses," *Academic Journal of Business, Administration, Law and Social Sciences*, vol. 1, p. 1, 2015.
- [2] Bughin et. al, "The impact of Internet technologies: Search," McKinsey Company, 2011.
- [3] APJII, "Asosiasi Penyelenggara Jasa Internet Indonesia. Penetrasi & Perilaku Pengguna Internet Indonesia. Asosiasi Penyelenggara Jasa Internet Indonesia (APJII)," APJII, 2016.
- [4] A. Almarabeh.T. , "A General Framework for E-Government: Definition Maturity Challenges, Opportunities, and Success," *European Journal of Scientific Research ISSN 1450-216X*, vol. 39 No. 1, no. 29-42, p. 1, 2010.
- [5] J. Fang, J. Wu and W. Tian, "On the Relationship between Internet and Traditional Communication Industry," pp. 1-6, 2017.
- [6] Z. Fang, E-Government in Digital Era: Concept, Practice, and Development, 2002.

- [7] L. A. ITS, "Egov Bench [Online]," [Online]. Available: <http://egovbench.addi.is.its.ac.id>.
- [8] Y. T. Z. J. L. C. H. C. S. C. H. C. H. C. W. .... C. Y. C. Liao, "Data adapter for querying and transformation between SQL and NoSQL database," *Future Generation Computer Systems*, 2016, p. 111–121.
- [9] M. T. Y. M. T. J. & C. R. González-Aparicio, "Testing of transactional services in NoSQL key-value databases.," 80, 384–399, *Future Generation Computer Systems*, 2018.
- [10] R. G. G. P. A. O. C. Györödi, "A comparative study: MongoDB vs. MySQL," in *13th International Conference on Engineering of Modern Electric Systems (EMES)*, page. 1-6, ISBN 978-1-4799-7650-8, Oradea, Romania, 2015.
- [11] O. d. N. Unidas., "E-Government Survey: e-Government for the Future We Want.," [https://doi.org/10.1016/S1369-7021\(02\)00629-6](https://doi.org/10.1016/S1369-7021(02)00629-6), United Nations, 2014.
- [12] S. S. a. G. Z. Yunhua Gu, "Application of NoSQL database in web crawling," *International Journal of Digital Content Technology and its Applications*, vol. 5, no. 6, p. 261–266, 2011.
- [13] Z. Majkic, "Intensional RDB Manifesto: a Unifying NewSQL Model for Flexible Big Data," 2014. [Online]. Available: <http://arxiv.org/abs/1403.0017>.
- [14] S. & N. G. Gupta, "Efficient Query Analysis and Performance Evaluation of the NoSQL Data Store for Big Data. In Proceedings of the First International Conference on Computational Intelligence and Informatics," in *In Proceedings of the First International Conference on Computational Intelligence and Informatics (pp. 549-558)*, Singapore, 2017.
- [15] R. M. Fifer, "Cost benchmarking functions in the value chain," *Planning Review* 17(3):18–19, 1989.
- [16] P. E. I. F. S. G. P. H. A. H. A. .... a. C.-M. P. E. I. F. S. G. P. H. A. H. A. .... a. M. W. (. W. (. Cudr'e-Mauroux1, "NoSQL database for RDF: An Empirical Evaluation, 310–311.," Retrieved from [http://ribs.csres.utexas.edu/nosqlrdf/nosqlrdf\\_iswc2013.pdf](http://ribs.csres.utexas.edu/nosqlrdf/nosqlrdf_iswc2013.pdf), 2013.
- [17] J. R. C. B. C. P. V. M. & B. J. Lourenço, "Choosing the right NoSQL database for the job: a quality attribute evaluation. *Journal of Big Data*, 2(1)," 2(1), 1–26. <https://doi.org/10.1186/s40537-015-0025->, 2015.