ISSN 2347 - 3983

Volume 9. No. 2, February 2021

International Journal of Emerging Trends in Engineering Research

Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter02922021.pdf

https://doi.org/10.30534/ijeter/2021/02922021

Designing Graph Database for Managing Lecturers Scientific Publications Using Neo4j

Muhammad Sholeh[1], Rr. Yuliana Rachmawati[2], Dina Andayati[3], Suraya[4]
[1,2]Department of Informatics, Faculty of Industrial Technology
Institut Sains & Teknologi AKPRIND Yogyakarta
[1]muhash@akprind.ac.id, [2]Yuliana@akprind.ac.id, [3]dina_asnawi@yahoo.com, [4]suraya@akprind.ac.id

ABSTRACT

This study aimed to implement a database design to manage lecturers' scientific publications using a graph database. Data of lecturers' scientific publications are very important to be managed properly. These data are much needed as the requirement of university accreditation, university clustering, and proposal of lecturers' functional positions. The increasing number of data certainly requires a good database design and one of the alternative databases is a graph database which is a model of a NoSQL database. This study was conducted in several stages: data collection and analysis, conceptual level database design, logical level database design, and physical level database design. This study developed a graph database design that stored various data about lecturers' scientific publications using Neo4j.

Key words : Publication, Scientific, Lecturer, Database, Graph.

1. INTRODUCTION

Data of lecturers' scientific publications as a portfolio of the three pillars of higher education (Education and Teaching, Research and development, Community service) implementation must be well managed. The lecturers' data are much needed both for the lecturers and the universities. For universities, the data are one of the requirements for university accreditation. For lecturers, the data are one of the requirements of the lecturers' workload and credit score assessment. To manage these data, it is necessary to develop an integrated information system.

The process of developing an information system requires a good and reliable system of database management. The function of an information system is only to present data stored in a database so that the process of database design must be done carefully to avoid data duplication. There are many database models and the database model that is widely used today is the SQL model or the structured database model. Besides the SQL database, there are other alternatives that are the NoSQL model or the unstructured database model. The NoSQL database is used for a specific data model and has a flexible scheme. Moreover, it uses a more dynamic database structure. By using the NoSQL database, problems in a relational database that are the problems of data volume, data speed, and data diversity can be resolved. [1]. The NoSQL database is divided into four categories that are a key-value database, a column family database, a document database, and a graph database. [2].

A graph database is the type of databases that uses a graph structure in the data storage process. A graph database does not have a schema like in a relational database that is tables and columns. The data will be stored in the form of nodes, properties, and edges. Figure 1 is a simple example of a graph database. In Figure 1, each circle represents a node. Each node can be related to other nodes through a line that has a direction and is called an edge. Both the nodes and the edges may have properties. [3]

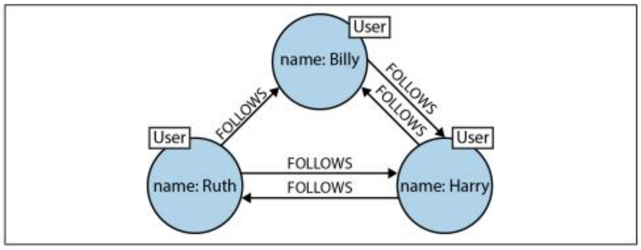
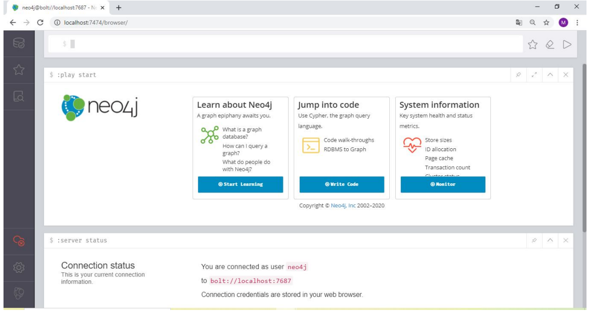


Figure 1. A graph database [3]

An application called Neo4j can be used to implement a graph database. By using Neo4j, the developed database can meet the ACID requirements (Atomicity, consistency, isolation, durability). ACID is the concept applied to a transactional database that is database that supports transactions [2]To run Neo4j, it can be done by visiting http://localhost:7474 through a browser. The front page display of Neo4j is shown in figure 2.



$:play start

Learn about Neo4j
A graph epiphany awaits you.
What is a graph database?
How can I query a graph?
What do people do with Neo4j?

Jump into code
Use Cypher, the graph query language.
Code walk-throughs
RDBMS to Graph

System information
Key system health and status metrics.
Store sizes
ID allocation
Page cache
Transaction count

Copyright © Neo4j, Inc 2002–2020

$:server status

Connection status
This is your current connection information.

You are connected as user neo4j
to bolt://localhost:7687
Connection credentials are stored in your web browser.

Figure 2. The front page display of Neo4j

There are several previous studies on databases. The study of Sholeh [4]resulted in a graph database for Micro, Small & Medium Enterprises (MSME) in Bantul District using Neo4j. The graph database was used to manage sales transactions. The implementation in the graph database consisted of nodes that stored data about buyers, products, and categories. To relate a node to other nodes, the researcher used edges two of which membeli_produk and jual_produk. Other studies that developed applications with graph databases were carried out by [5], [6], [7], [8], [9], [10]

The study of Patil [11] investigated the importance of data analysis, data management, and big data in social and business perspectives. A graph database is more suitable to handle dynamic and complex data in big and unstructured data. Each element of data is represented by a node. For example, in the database of a social media site a person was represented by a node, and the properties are name, age, likes and dislikes, etc. The nodes are related by edges. The use of graph database is expected to be useful in process involving data. Big data processing certainly needed the right and efficient computational technique.

Vyawahare [12] states that a relational database model is one of the databases using structured data. Currently, the growth in use of internet especially social networking gives an impact on the demands of big data and unstructured data. One of the databases that can be used to handle unstructured data is a graph database (NoSQL database). Due to the limitations having by a relational database in handling data relationships (highly connected data), a graph database can be used as an alternative in handling big data. In the query form and data properties, a relational database and a graph database have their advantages and disadvantages. In the information system developed by Vyawahare, the data were stored in a relational database so that the process of application conversion from a relational format to a graph could be very useful. This study was conducted to develop a dual database system through migration which integrated the advantages of a relational database and a graph database. This study showed that this hybrid system performed efficiently.

Wiktorski [13] states that the technology of a graph database contains several technological features that is applicable to SQL database, such as the ACID property and the search process. There are some applications to implement a graph database such as Neo4j, OrientDB, InfiniteGraph, FlockDB, AllegroGraph, and etc. In the era of big data, problems related to database are scalability for big data and scaling for read / write operations. The study of Wiktorski discussed the progress and limitations in managing graph database.

Medhi [14]investigated the use of a relational model database which is widely used in developing information systems. A relational database model was used to store, manipulate, and retrieve data. The use of a relational database is decreasing in use due to the growth of big data. One of the solutions is to use a graph database to store data. A graph database can be used to store big and unstructured data.

Studies conducted by Wirawan [15] [16] investigated a graph database to present information about corridors for passengers of Bus Rapid Transit (BRT). A graph database was used to represent BRT because a graph could show nodes and relationships. This study investigated the implementation of a graph database for data used in BRT and designed a travel search algorithm for BRT corridors. Algorithms were used to develop the application that utilizes data with a graph database. With the application that used the graph database, it showed that the graph database could be used for BRT data and the algorithm used could present BRT route information and at the same time could convey information in the process of corridor movement.

Other studies that investigated the use of a graph database are [17], [18],[19],[20],[21], [22]

Based on the literature review, this study aimed to implement a database design to manage lecturers' scientific publications using a graph database. The nodes stored data such as lecturers' identities, scientific publications in journals and proceedings, Intellectual Property Rights (IPR), reports of community service, journals, and lecturers' research.

2. MATERIALS AND METHOD

A. The Conceptual Design of the Graph Database

The first step in the graph database design is to analyze and collect data, especially the data to be stored in the database. Data were grouped by type into publications in journals, seminars, IPR, community service and etc. The results of this data collection then were used to design a graph database. Data were made in a circle that represented a node and each node had properties. The researchers created a lecturer node which was related to the nodes of journals, seminars, intellectual property rights (IPR) and etc. Each node contained data; for example, the lecturer node contained properties of the lecturers' name, National Lecturer Identification Number (NIDN) and etc. The conceptual design of lecturers' scientific publications management using a graph database is shown in figure 3.

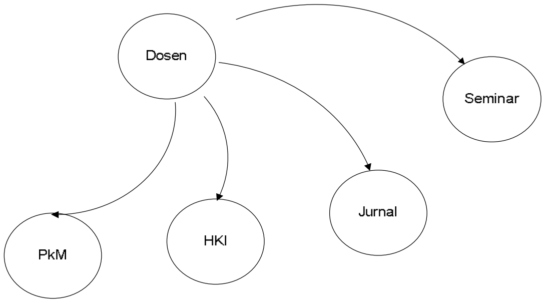


Figure 3. The conceptual design of lecturers' scientific publications management using a graph database

B. The Implementation the graph database using Neo4j

The results of the graph database analysis and design were implemented using Neo4j. Neo4j was run by visiting http://localhost:7474 through a browser. Before running this page, the researchers activated Neo4j. The basic commands during the database management process were the operations of *create, retrieve, update and delete* (CRUD). The graph creation process began with making the nodes and then relating the nodes to each other. The figure 4 is an example of the result of creating person nodes and relating them in a friend relationship [23].

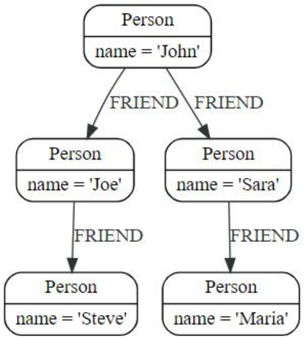
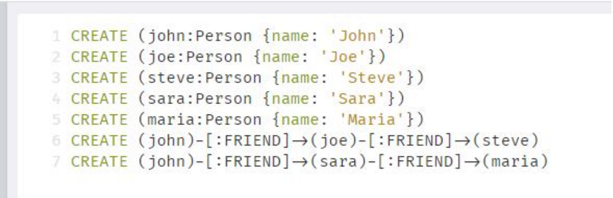


Figure 4. An example of a relationship between nodes (Neo4j 2020).

The command for making the nodes and edges is shown in Figure 5.



```
CREATE (john:Person {name: 'John'})
CREATE (joe:Person {name: 'Joe'})
CREATE (steve:Person {name: 'Steve'})
CREATE (sara:Person {name: 'Sara'})
CREATE (maria:Person {name: 'Maria'})
CREATE (john)-[:FRIEND]→(joe)-[:FRIEND]→(steve)
CREATE (john)-[:FRIEND]→(sara)-[:FRIEND]→(maria)
```

Figure 5. Making the person nodes

3. RESULTS AND DISCUSSION

A.Making the Nodes

Based on the analysis and design, the nodes that were made were:

- The lecturer node contained properties about the lecturers' identities
- The article node contained properties about the journal articles
- The seminar node contained properties about the proceeding papers
- The IPR node contained properties about the intellectual property rights
- The community service node contained properties about the reports of community service
- The journal node contained properties about the names of journals where the lecturers published their articles
- The research node contained properties about lecturers' research

B. Making the Edges

The researcher made the nodes and then related them to the other nodes with edges. With these edges, the data were presented in more detail. The edges that were made were:

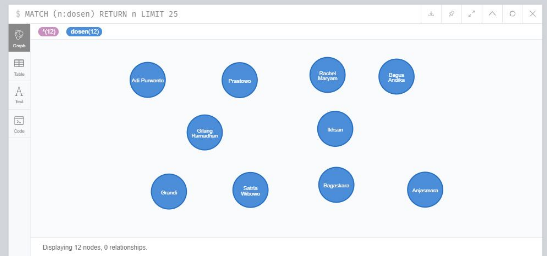
- The edge of MELAKSANAKAN_ABDIMAS was used to relate the lecturer node with the community service node
- The edge of MENGHASILKAN_KARYA_HKI was used to relate the lecturer node with the IPR node
- The edge of MENGHASILKAN_KARYA_JURNAL was used to relate the lecturer node with the article node
- The edge of MENGHASILKAN_MAKALAH_SEMINAR was used to relate the lecturer node with the seminar node
- The edge of DIPUBLIKASIKAN_PADA_JURNAL was used to relate the article node with the journal node

C. Implementation in Neo4j

1. Making the lecturer node. This node stored data about the lecturers' identities. The required properties were the NIDN and the lecturers' name. The commands to make the lecturer node:

   $ create(:dosen {nidn:"0510027001",nama: "Bagaskara"}):
   $ create(:dosen {nidn:"0510036801",nama: "Anjasmara"})
   $ create(:dosen {nidn:"0523037901",nama: "Ikhsan"})
   $ create(:dosen {nidn:"05300xx01",nama: "Muhammad Sholeh"})

   The command used to see the contents in the lecturer node:
   $ MATCH (n:dosen) RETURN n
   The results are in figure 6.



```
$ MATCH (n:dosen) RETURN n LIMIT 25
```

Displaying 12 nodes, 0 relationships.

Figure 6. The results of the lecturer node in a graph form

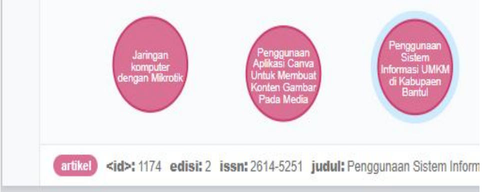
2. Making the article node. This node stored data about the journal articles. The commands to make the article node:

```
$ create (:artikel{nidn:" 0510027001",judul:"
  Penggunaan Sistem Informasi UMKM di
  Kabupaten Bantul",volume:3, edisi:2,tahun:”
  November 2019,issn:"2614-5251",status:”
  Terakreditasi Sinta 5”})
$ create
  (:artikel{nidn:"05300xx01",judul:"Penggunaan
  Aplikasi Canva Untuk Membuat Konten Gambar
  Pada Media Sosial Sebagai Upaya
  Mempromosikan Hasil Produk UKM", volume:4,
  edisi:1,tahun:"November
  2020",issn:"2614-5251",status:"Terakreditasi
  Sinta 5"})
```

The command used to see the contents in the article node:

```
$ MATCH (n:artikel) RETURN n
```

The results are in Figure 7.


Jaringan komputer dengan Mikrotik
Penggunaan Aplikasi Canva Untuk Membuat Konten Gambar Pada Media
Penggunaan Sistem Informasi UMKM di Kabupaen Bantul

artikel <id>: 1174 edisi: 2 issn: 2614-5251 judul: Penggunaan Sistem Inform

Figure 7. The results of the article node in a graph form

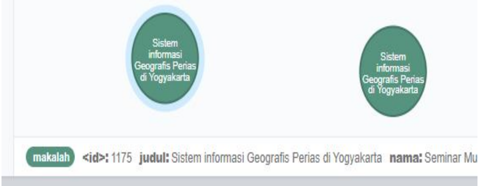
3. Making the seminar node. This node stored data about the proceeding papers. The commands to make the seminar node:

```
$create (:makalah{nidn:"05300xx01",judul:"Sistem
informasi Geografis Perias di Yogyakarta", nama:"
SMAI ", penyelenggara:”Universitas Mercu Buana
Yogyakarta”, tahun:2020"})
```

The command used to see the contents in the seminar node:

```
$ MATCH (n:makalah) RETURN n
```

The results are in Figure 8.


Sistem informasi Geografis Perias di Yogyakarta
Sistem informasi Geografis Perias di Yogyakarta

makalah <id>: 1175 judul: Sistem informasi Geografis Perias di Yogyakarta nama: Seminar Mu

Figure 8. The results of the seminar node in a graph form

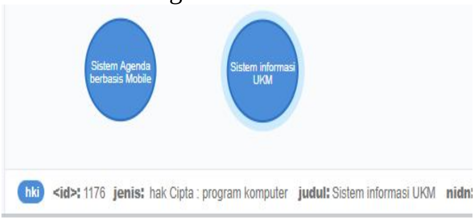
4. Making the IPR node. This node stored data related to the lecturers' intellectual property rights. The required properties were the NIDN, the IPR titles, and the types of copyright. The commands to make the IPR node:

```
$ create (:hki{nidn:"05300xx01",judul:"Sistem
informasi UKM ", jenis:" hak Cipta : program
komputer ", no:”E000001 ”,tahun:2020})
```

The command used to see the contents in the IPR node:

```
$ MATCH (n:hki) RETURN n
```

The results are in figure 9.



hki <id>: 1176 jenis: hak Cipta : program komputer judul: Sistem informasi UKM nidn:

Figure 9. The results of the IPR node in a graph form

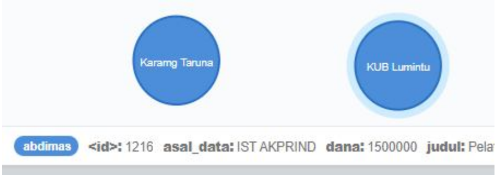
5. Making the community service node. This node stored data about the reports of community service activities. The required properties were the NIDN and the title of the community service activities. The commands to make the community service node:

```
$ create
(:abdimas{nidn:"05300xx01",judul:"Pelatihan
Media sosial”  , tempat:”KUB
Lumintu”,kota:”Bantul”, waktu:"11-12-2020”,
skema:”abdimas mandiri”,  , asal_data:”IST
AKPRIND”,dana:1500000 ,tahun:2020})
```

The command used to see the contents of the community service node:

```
$ MATCH (n”abdimas) RETURN n
```

The results are in figure 10.



abdimas <id>: 1216 asal_data: IST AKPRIND dana: 1500000 judul: Pela

Figure 10. The results of the community service node in a graph form

6. Making the journal node. This node stored data about the journals in which the lecturers’ published their articles. The required properties were International Standard Serial Number (ISSN) and the journal name. The commands to make the journal node:

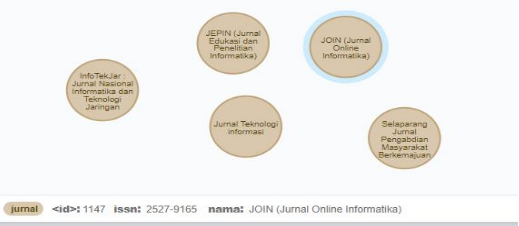
```
$ create(:jurnal{ nama:" Selaparang Jurnal
  Pengabdian Masyarakat
  Berkemajuan",issn:"2614-5251"})
```

```
$ create(:jurnal{ nama:" E-dimas: Jurnal
  Pengabdian Kepada
  Masyarakat",issn:"2087-3565"})
$ create(:jurnal{ nama:" Jurnal Penelitian Dan
  Pengabdian Kepada Masyarakat UNSIQ",issn:"
  2354-869x "})
$ create(:jurnal{ nama:" Jurnal Penelitian Dan
  Pengabdian Kepada Masyarakat UNSIQ",issn:"
  2354-869x "})
```

The command used to see the contents of the journal node:

```
$ MATCH (n:jurnal RETURN n
```

The results are in Figure 10.


JEPIN (Jurnal Edukasi dan Penelitian Informatika)
JOIN (Jurnal Online Informatika)
InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan
Jurnal Teknologi informasi
Selaparang Jurnal Pengabdian Masyarakat Berkemajuan

```
jurnal  <id>: 1147  issn: 2527-9165  nama: JOIN (Jurnal Online Informatika)
```

Figure 10. The results of the journal node in a graph form

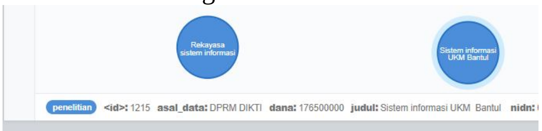
7. Making the research node. This node stored data related to lecturers' research. The required properties consisted of the ISSN and the journal name. The commands to make the research node:

```
$ create
(:penelitian{nidn:"05300xx01",judul:"Sistem
informasi UKM  Bantul ", skema:" PTUPT ",
asal_data:”DPRM DIKTI”,dana:176500000
,tahun:2020})
```

The command used to see the contents of the research node:

```
$ MATCH (n: penelitian)  RETURN n
```

The results are in figure 11.



```
penelitian  <id>: 1215  asal_data: DPRM DIKTI  dana: 176500000  judul: Sistem informasi UKM  Bantul  nidn:
```

Figure 11. The results of the research node in a graph form

8. Making the edge of MELAKSANAKAN_ABDIMAS. This edge was used to relate the lecturer node with the community service node.

```
$ match (d:dosen),(h:abdimas)
where (d.nidn="05300xx01" and
h.nidn="05300xx01")
CREATE
(d)-[:MELAKSANAKAN_ABDIMAS] -> (h)
```

The commands used to relate the lecturer node with the community service node, especially the lecturers who had NIDN 05300xx01:

9. Making the edge of PENGHASILKAN_KARYA_HKI. This edge was used to relate the lecturer node with the IPR node.

```
$match (d:dosen),(h:hki)
where (d.nidn="05300xx01" and
h.nidn="05300xx01")
CREATE
(d)-[:MENGHASILKAN_KARYA_HKI -> (h)
```

The commands used to relate the lecturer node with the IPR node, especially the lecturers who had NIDN 05300xx01:

10. Making the edge of PENGHASILKAN_KARYA_JURNAL. This edge was used to relate the lecturer node with the article node.

```
$match (a:artikel),(d:dosen)
where a.nidn="0510027001" AND
d.nidn="0510027001"
CREATE
(d)-[:MENGHASILKAN_KARYA_JURN
AL]  -> (a)
```

The commands used to relate the lecturer node with the article node, especially the lecturers who had NIDN 0510027001:

11. Making the edge of BERHASILKAN_MAKALAH_SEMINAR. This edge was used to relate the lecturer node with the article node.

```
$ match (d:dosen),(m:makalah)
where (d.nidn="05300xx01" and
m.nidn="05300xx01")
CREATE
(d)-[:MENGHASILKAN_MAKALAH_SEMIN
AR] -> (m)
```

The commands used to relate the lecturer node and the article node, especially the lecturers who had NIDN 05300xx01:

12. Making the edge of MENGHASILKAN_PENELITIAN. This edge was used to relate the lecturer node with the research node.

```
$ match (d:dosen),(h:penelitian)
where (d.nidn="05300xx01" and
h.nidn="05300xx01")
CREATE
(d)-[:MENGHASILKAN_PENELITIAN] -> (h)
```

The commands used to relate the lecturer node with the research node, especially the lecturers who had NIDN 05300xx01:

13. Making the edge of DIPUBLIKASIKAN_PADA_JURNAL. This edge

was used to relate the lecturer node with the article node and the journal node. These related nodes presented the data of lecturers who had made scientific publications and the names of journals. The results in Figure 9 did not have journal identity information and this node presented detailed information.

```
$ match (d:dosen),(a:artikel),(j:jurnal)
  where (d.nidn=" 0510027001" and a.nidn=" 0510027001")  AND (a.issn="2614-5251" AND j.issn="2614-5251")
    CREATE
(a)-[:DIPUBLIKASIKAN_PADA_JURNAL] -> (j)
```

This command was used to relate between the lecturer node with the article node and the journal node, especially lecturers who had NIDN 0510027001 and published their articles in journals with ISSN 2614-5251.

D. Results of the relationship between nodes

From the data included in the graph database, the relationships between nodes could be simulated. The simulation results were

1. Displays all vertices and the relationship between vertices.

   The figure 12 shows the lecturer node which is connected with all nodes. Lecturer nodes are linked with IPR nodes, abdimas, research, papers, articles. Specifically, article nodes are linked to journal nodes.

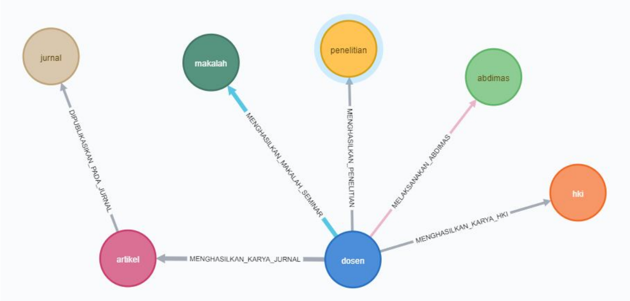
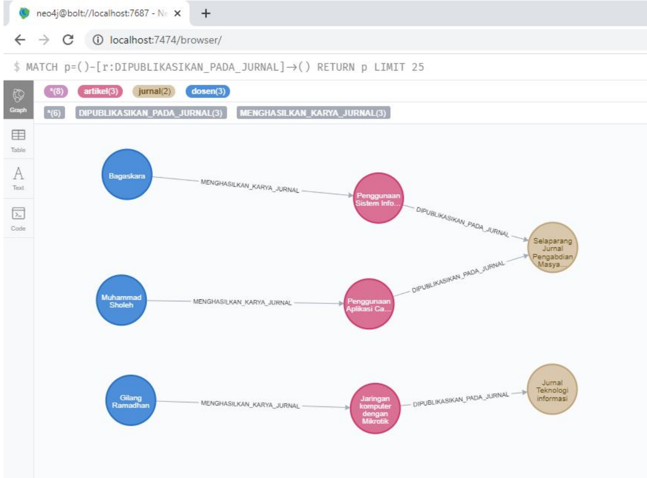


Figure 12 Links between the nodes

2. The data of lecturers who made scientific publications in journals.

   The figure 13 presents that there are 3 lecturers each of whom made a scientific publication in 2 journals.

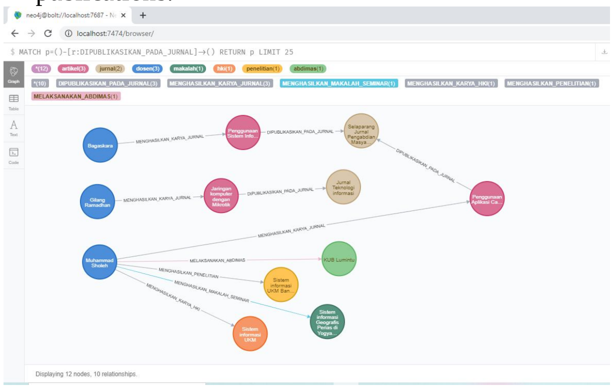

neo4j@bolt://localhost:7687 - Ne
localhost:7474/browser/
Graph
Table
Text
Code
*(8)
artikel(3)
jurnal(2)
dosen(3)
*(6)
DIPUBLIKASIKAN_PADA_JURNAL(3)
MENGHASILKAN_KARYA_JURNAL(3)
Bagaskara
Muhammad Sholeh
Gilang Ramadhan
Penggunaan Sistem Info...
Penggunaan Aplikasi Ca...
Jaringan komputer dengan Mikrotik
Selaparang Jurnal Pengabdian Masya...
Jurnal Teknologi informasi
MENGHASILKAN_KARYA_JURNAL
MENGHASILKAN_KARYA_JURNAL
MENGHASILKAN_KARYA_JURNAL
DIPUBLIKASIKAN_PADA_JURNAL
DIPUBLIKASIKAN_PADA_JURNAL
DIPUBLIKASIKAN_PADA_JURNAL

```
$ MATCH p=()-[r:DIPUBLIKASIKAN_PADA_JURNAL]→() RETURN p LIMIT 25
```

Figure 13. The data of the lecturers who made scientific publications.

3. The data of the lecturers' publications.

   The figure 14 presents the data of the lecturers' publications.


neo4j@bolt://localhost:7687 - Ne
localhost:7474/browser/
Graph
Table
Text
Code
*(12)
artikel(3)
jurnal(2)
dosen(3)
makalah(1)
hki(1)
penelitian(1)
abdimas(1)
*(10)
DIPUBLIKASIKAN_PADA_JURNAL(3)
MENGHASILKAN_KARYA_JURNAL(3)
MENGHASILKAN_MAKALAH_SEMINAR(1)
MENGHASILKAN_KARYA_HKI(1)
MENGHASILKAN_PENELITIAN(1)
MELAKSANAKAN_ABDIMAS(1)
Bagaskara
Gilang Ramadhan
Muhammad Sholeh
Penggunaan Sistem Info...
Jaringan komputer dengan Mikrotik
Penggunaan Aplikasi Ca...
Selaparang Jurnal Pengabdian Masya...
Jurnal Teknologi informasi
KUB Lumintu
Sistem informasi UKM Ban...
Sistem informasi Geografis Perias di Yogya...
Sistem informasi UKM
MENGHASILKAN_KARYA_JURNAL
DIPUBLIKASIKAN_PADA_JURNAL
MELAKSANAKAN_ABDIMAS
MENGHASILKAN_PENELITIAN
MENGHASILKAN_MAKALAH_SEMINAR
MENGHASILKAN_KARYA_HKI

```
$ MATCH p=()-[r:DIPUBLIKASIKAN_PADA_JURNAL]→() RETURN p LIMIT 25
```

Displaying 12 nodes, 10 relationships.

Figure 14. The data of the lecturers' publications.

4. CONCLUSION

The use of graph database is very suitable for unstructured data. The process of completing data can be customized to the needs and there must still be a minimum of data that must be included. This study on the database design for the management of lecturers' scientific publications is designed with several nodes and different data completeness. The design and implementation results can simulate the data stored in a graph.

REFERENCES

[1] B. M. Sasaki, J. Chao, and R. Howard, *Graph Databases For Beginners*. neo4j.com, 2018.

[2] A. Kadir, *Dasar Basis Data Graft & NEo4j*. Yogyakarta: Penerbit Andi, 2020.

[3] I. Robinson, J. Webber, and E. Eifrem, *Graph Databases New Opportunities For Connected Data*. neo4j.com, 2015.

[4] M. Sholeh, 'Pemodelan Basis data Graph dengan

Neo4j (Studi Kasus : Basis Data Sistem Informasi Penjualan pada UMKM)', *Jurnal Teknologi Informasi dan Terapan (J-TIT) Vol.*, vol. 7, no. 1, pp. 25–32, 2020.

[5] C. M. Widyayanti and A. I. Nurhidayat, 'RANCANG BANGUN SISTEM INFORMASI HOTEL MENGGUNAKAN GRAPH DATABASE', *Jurnal Manjemen Informatika*, vol. 6, no. 2, pp. 14–22, 2016.

[6] Y. Y. Sahria and D. H. Fudholi, 'Pemodelan Pengetahuan Graph Database Untuk Jejaring Penelitian Kesehatan di Indonesia', *Jurnal Media Informatika Budidarma*, vol. 4, no. 3, pp. 604–611, 2020, doi: 10.30865/mib.v4i3.2183.

[7] A. F. Efendi, T. Setiadi, T. Informatika, F. T. Industri, and U. A. Dahlan, 'PENERAPAN GRAPH DATABASE UNTUK SISTEM REKOMENDASI KOLEKSI PERPUSTAKAAN DI ERA BIG', *Journal of Documentation and Information Science*, vol. 3, no. 2, pp. 143–150, 2019.

[8] M. B. Rakesh and S. B. N. Kumar, 'Implementation of Movie Recommender System Based on Neo4J Graph Database', vol. 6, no. 2, pp. 49–52, 2019.

[9] E. Zhang, J. Fiaidhi, S. Mohammed, O. Rd, T. Bay, and O. N. Pb, 'Social Recommendation using Graph Database Neo4j ： Mini Blog , Twitter Social Network Graph Case Study', *International Journal of Future Generation Communication and Networking*, vol. 10, no. 2, pp. 9–20, 2017.

[10] S. Medhi, 'Visualization of Graph Models for Web Document in Neo4j', vol. 7, no. 5, pp. 13–20, 2016.

[11] N. S. Patil, P. Kiran, N. P. Kavya, and K. M. Naresh Patel, 'A survey on graph database management techniques for huge unstructured data', *International Journal of Electrical and Computer Engineering*, vol. 8, no. 2, pp. 1140–1149, 2018, doi: 10.11591/ijece.v8i2.pp1140-1149.

[12] H. R. Vyawahare, P. P. Karde, and V. M. Thakare, 'Improving performance using relational and graph database', *International Journal of Recent Technology and Engineering*, vol. 8, no. 2, pp. 1722–1726, 2019, doi: 10.35940/ijrte.B1862.078219.

[13] C. Strauch and W. Kriha, *NOSQL databases*. 2019.

[14] S. Medhi and H. Baruah, 'Relational database and graph database: A comparative analysis', *Journal of Process Management. New Technologies*, vol. 5, no. 2, pp. 1–9, 2017, doi: 10.5937/jouproman5-13553.

[15] P. W. Wirawan and D. E. Riyanto, 'Kajian Implementasi Graph Database pada Rute Bus Rapid Transit', *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 3, no. 3, pp. 313–319, 2017, doi: 10.25077/teknosi.v3i3.2017.313-319.

[16] P. W. Wirawan, D. E. Riyanto, D. Mutiara, and K. Nugraheni, 'Graph Database Schema for Multimodal Transportation in Semarang', *Journal of Information Systems Engineering and Business Intelligence*, vol. 5, no. 2, pp. 163–170, 2019.

[17] J. Pokorný, M. Valenta, and M. Troup, 'Indexing Patterns in Graph Databases', in *Proceedings ofthe 7th International Conference on Data Science, Technology and Applications (DATA 2018)*, 2018, pp. 313–321, doi: 10.5220/0006826903130321.

[18] W. Silva, S. Lifschitz, and H. Vera, 'Designing Graph Databases With GRAPHED', *Journal of Database Management*, vol. 30, no. 1, p. 41, 2019, doi: 10.4018/JDM.2019010103.

[19] F. Gong, Y. M. Id, W. Gong, X. Li, and C. L. Id, 'Neo4j graph database realizes efficient storage performance of oilfield ontology', pp. 1–16, 2018.

[20] F. M. S. Lopez and E. G. S. D. La Cruz, 'Literature review about Neo4j graph database as a feasible alternative for replacing RDBMS', *Revista*, vol. 18, no. 2, pp. 135–139, 2015.

[21] M. M. Dagar, P. Aitm, M. Singh, and A. Palwal, 'REDUCING TIME COMPLEXITY OF RELATIONAL DATABASES USING NEO4J', *International Journal of Innovative Research in Technology & Science(IJIRTS)*, vol. 1, no. 2, pp. 1–6, 2019.

[22] J. Pokorny, 'Modelling of Graph Databases', pp. 4–17, 2017.

[23] Neo4j, 'The Neo4j Cypher Manual 3.5 Online]', 2020. [Online]. Available: https://neo4j.com/docs/cypher-manual/3.5/introduction/.