



International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

Prescription of Food and Medicines Using Neo4j

Siddhesh Inamdar¹, Pranav Sahu², Yash Dholakia³

U.G Student, Dept of I.T, Rajiv Gandhi Institute of Technology, Mumbai, India¹

U.G Student, Dept of I.T, Rajiv Gandhi Institute of Technology, Mumbai, India²

U.G Student, Dept of I.T, Rajiv Gandhi Institute of Technology, Mumbai, India³

ABSTRACT: This paper is regarding opinion and decision making system (prescription) which helps a doctor prescribe medicines to a patient. The doctor is able to prescribe medicines only after determining what foods and medicines the patient is allergic to. Different foods and medicines contain specific components to which the patient may be allergic. Different patients are allergic to different components in both food and medicine. Sometimes these allergic reactions could be fatal. So while prescribing medicines to patients, the doctor has to ensure that the prescribed medicine does not contain those components to which the patient is allergic to. This paper provides details about how a graphical representation of data can be used by the doctor to analyse an individual patient's records and prescribe medicines according to his individual needs.

KEYWORDS: Neo4j, Create, Return, Doctor, Patient, Allergy, Component.

I. INTRODUCTION

The new way of representing database is introduced by NEO4J. When compared with traditional relational database, it has different advantages like faster processing for data relationships, flexibility of data model, zero latency and real-time performance, data and relationships stored together thereby improving the performance [1]. In sql, we have entity and in NEO4J we have nodes. These are main data elements. Different nodes have different properties and different nodes can be connected via relationships. Properties are attributes of nodes and they can be indexed and constrained. Labels are used to identify the nodes and these labels are unique because they are used to identify the nodes which are unique [2]. The syntax of creating the node is "CREATE (label:node { property1: "value1"})". The example used in our implementation is "CREATE (p:Patient { name: "Patient"})". Similarly, the syntax of creating the relationship between 2 nodes is "(label1)-[:rel-name]->(label2)". This can be exemplified using the example mentioned in our implementation "(p)-[:NAME]->(a)". Here, we draw the relationship NAME from node with label p to node with label a. These syntaxes are represented using Cypher.

II. LITERATURE SURVEY

Fraud detection can be done using NEO4J database graphical representation. Augmenting one's existing fraud detection infrastructure to support ring detection can be done by running appropriate entity link analysis queries using a graph database, and running checks during key stages in the customer & account lifecycle, such as:

At the time the account is created

During an investigation

As soon as a credit balance threshold is hit

When a check is bounced

Real-time graph traversals tied to the right kinds of events can help banks identify probable fraud rings: during or even before the Bust-Out occurs.

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

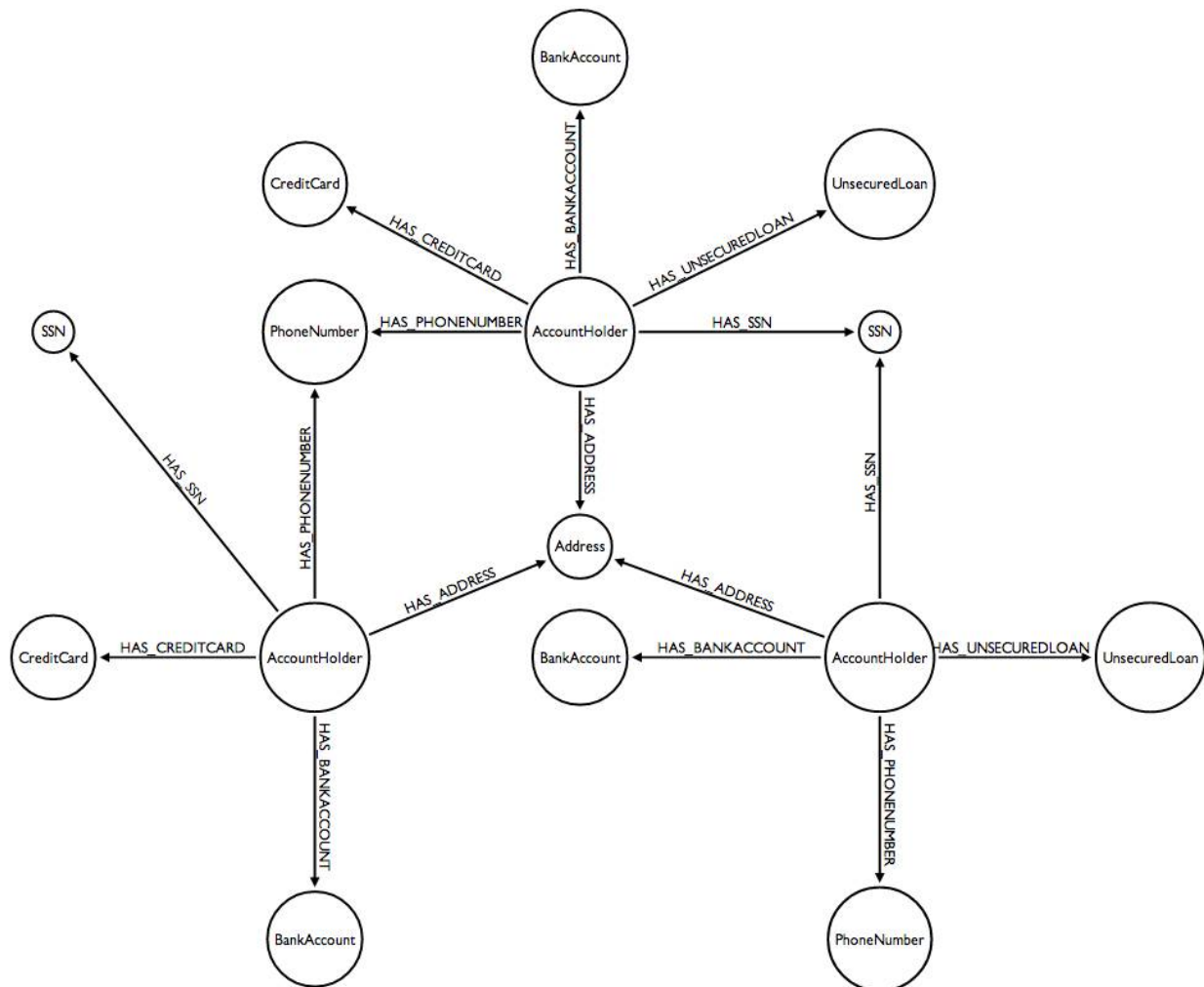


Fig 2.1 Fraud Detection using NEO4J

Thus, using NEO4J database graphical representation, fraud detection can be performed efficiently [3].

It can be used to track roles, groups and assets. Managing multiple changing roles, groups, products and authorizations is an increasingly complex task at your enterprise organization. Using Neo4j, you seamlessly track all identity and access authorizations and inheritances with substantial depth and real-time results. This interconnected view of your data gives better real-time insights and controls than ever before. Neo4j connects your data along intuitive relationships so identity and access management happens quickly and effectively [4].

It can also be used to augment the knowledge graph using highly contextual search results. Managing your organization's growing library of digital assets requires a highly contextual search solution. With Neo4j, you augment your enterprise search capabilities with knowledge graph i.e. *graph-based search* capabilities to deliver only relevant results. For example, a simple keyword search can be augmented with additional results that are related to the keyword without being explicitly requested in the search. Neo4j based graph-based search is used by companies to improve search capabilities of product, services, content and knowledge catalogues [5].



International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

III. PROPOSED SYSTEM

An opinion / decision system which helped doctors prescribe medicines according to individual patients need. A patient could be allergic to a compound in a food product or a medicine pill. Hence, prescribing medicines or a diet which could contain compounds that could be allergic to a patient which could be adverse. This database consists of number of patients and each and every food and medicine the patient is allergic to. This helps the doctor to prescribe accordingly. Written in “Cypher” language, the database is represented in graphical form which is very user friendly and easily understandable. The root node is a patient. Nodes are connected with arrows which describe the relations between them. This root node can further be extended which gives all the patients in the database, these individual patients can be extended which gives the nodes of Food and Medicines. These nodes can further be extended to give the names to which the patient is allergic to, which helps the doctor to prescribe medicines without these compounds. Neo4j is much faster, easily understandable and can save huge quantity of data and process queries much faster than the traditional databases, making it an ideal platform.

Block Diagram:

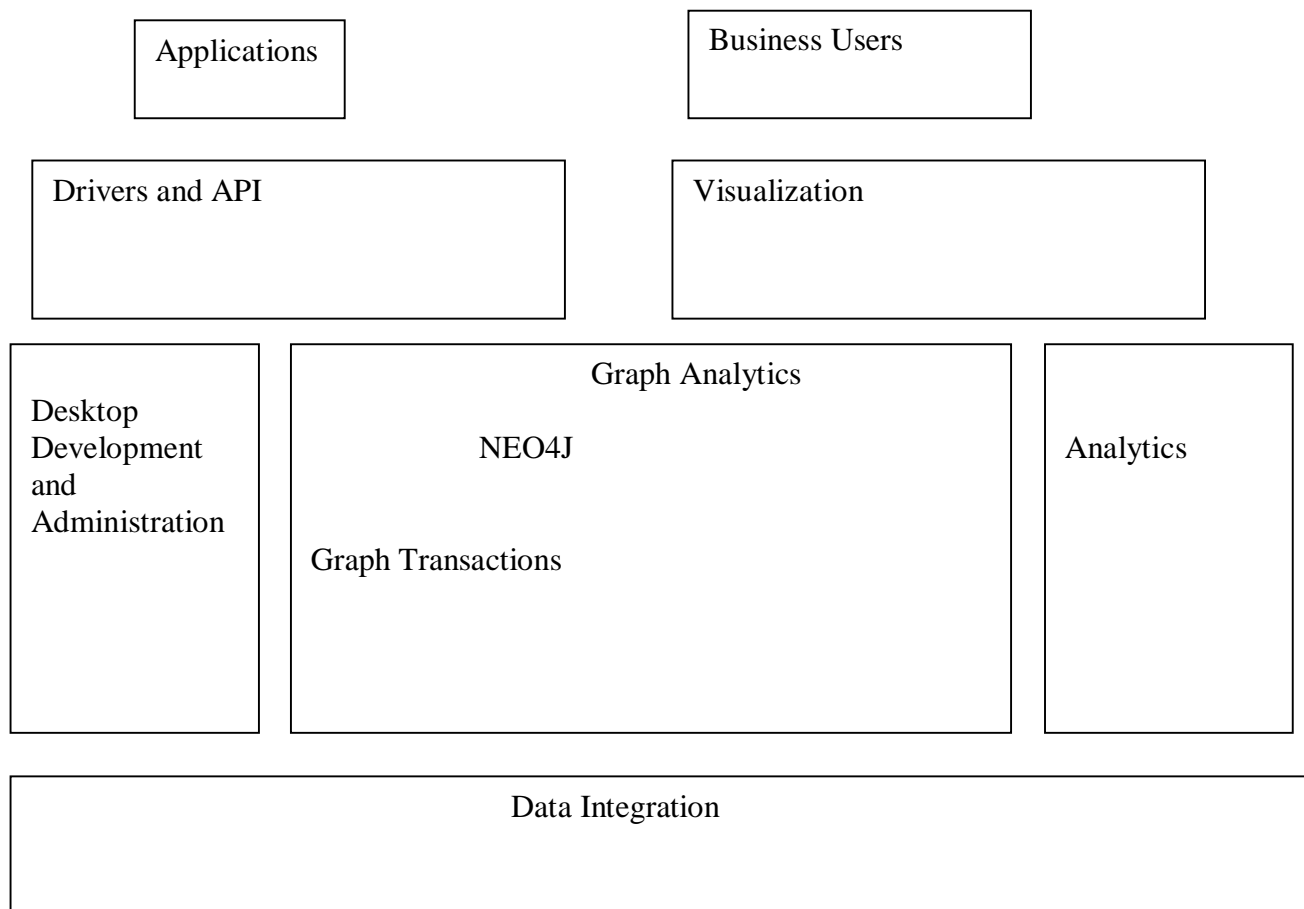


Fig 3.1 Block Diagram of NEO4J architecture

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

After referring the NEO4J site and its uses, this block diagram can be inferred. At the centre, there is NEO4J. It has several applications. It can be used for graph transactions and analytics. Graph transactions are used to add or remove nodes or relationships. Admin takes care of this graph transaction. Graph Analytics are used to analyse the data and make suitable inferences and they are used by data analysts and scientists. Drivers and API are used to make application usable by users. Visualization is done by business users to ensure that users are able to view the graph accurately as per suggested by business user. Thus, the most important thing is to generate the graph. The steps for generating graph are as follows.

Block Diagram for generating graph:

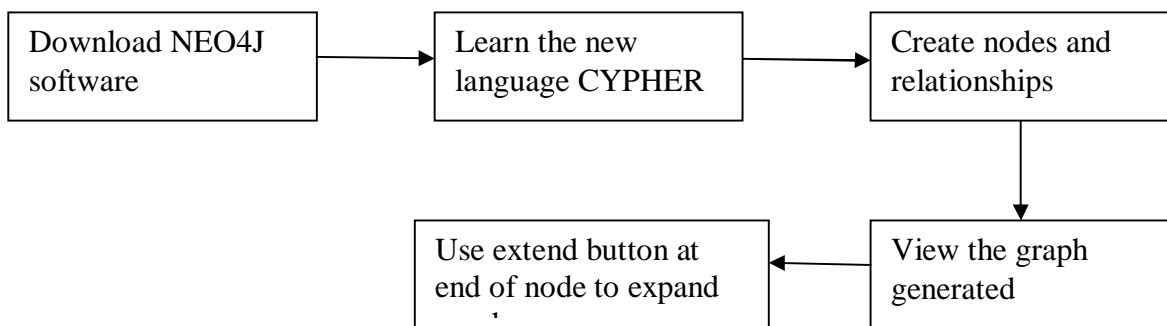


Fig 3.2 How to create the graph using NEO4J

ALGORITHM:

1. Create the node named 'patient' with label 'p'
2. Create the node named 'John' with label 'a'
3. Create the node named 'Perry' with label 'b'
4. Create the node named 'Teresa' with label 'c'
5. Create the node named 'Zee' with label 'd'
6. Create the node named 'Jason' with label 'e'
7. Create the node named 'Nick' with label 'f'
8. Create the node named 'Savannah' with label 'g'
9. Create the node named 'Tim' with label 'h'
10. Create the node named 'Han' with label 'i'
11. Create the node named 'Lee' with label 'j'
12. Create the node named 'Sam' with label 'k'
13. Create the node named 'Rashid' with label 'l'
14. Create the node named 'Tina' with label 'm'
15. Create the node named 'Abdul' with label 'n'
16. Create the node named 'Dan' with label 'o'
17. Create the relationship from patient node to each patient mentioned above using labels from node patient labelled 'p' to node with name of patient.
18. Create the nodes food and medicine for each patient.
19. Create the relationship from patient name node to food and medicine using labels from node patient to food and medicine.
20. Repeat step 21 for each patient name labelled as 'a' to 'o'
21. For each patient create node of food types and medicine types a patient is allergic to and assign a unique label to each node
22. Create the relationship from food node to food type patient is allergic to using appropriate label name.
23. Create the relationship from medicine node to medicine type patient is allergic to using appropriate label name.

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

24. Graph is generated and then use extend button at end of each node to extend the graph.

IV. RESULTS

We have written the code in “Cypher” language and made the database with this language. The database can be represented in graphical form and hence can be analysed more clearly and concisely. The functionality for showing connected nodes is called as extend. We have initially created a single node called patient.



Fig 4.1 Create node patient

Then, if we extend the patient node, we get all the patients connected to this patient node. The relationship is name.

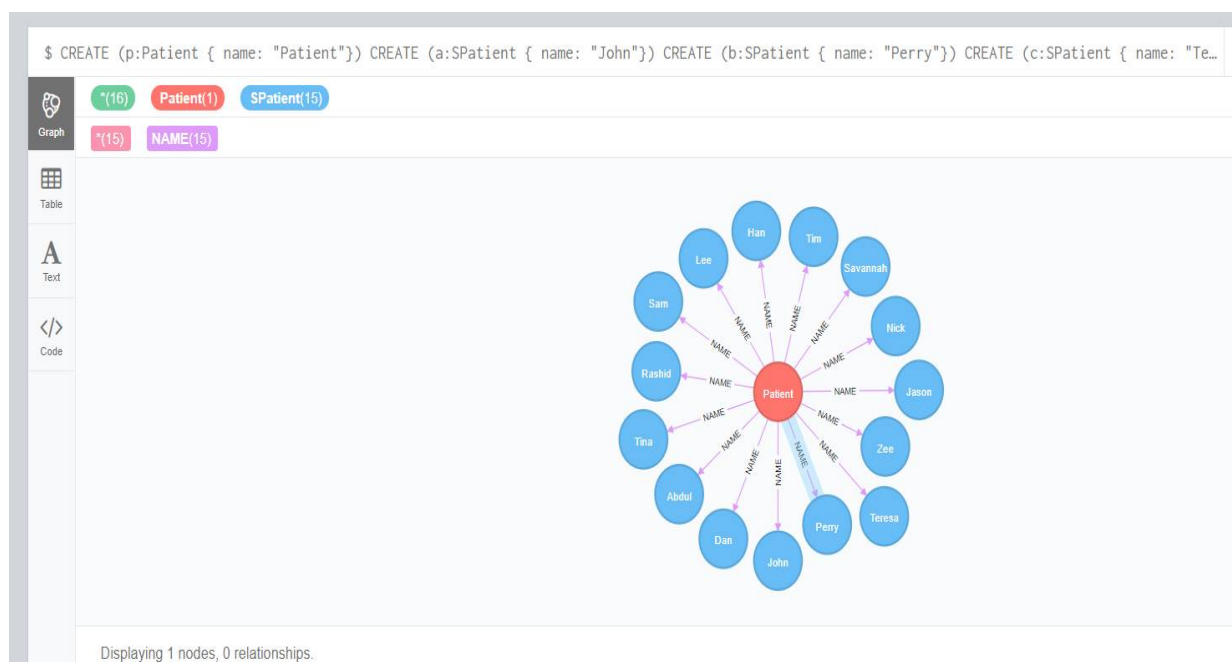


Fig 4.2 Extend Patient node

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijirccce.com

Vol. 5, Issue 10, October 2017

Now, if we click on any patient's name and extend it, we see the nodes food and medicine that patient is allergic to using the relationship allergic to.

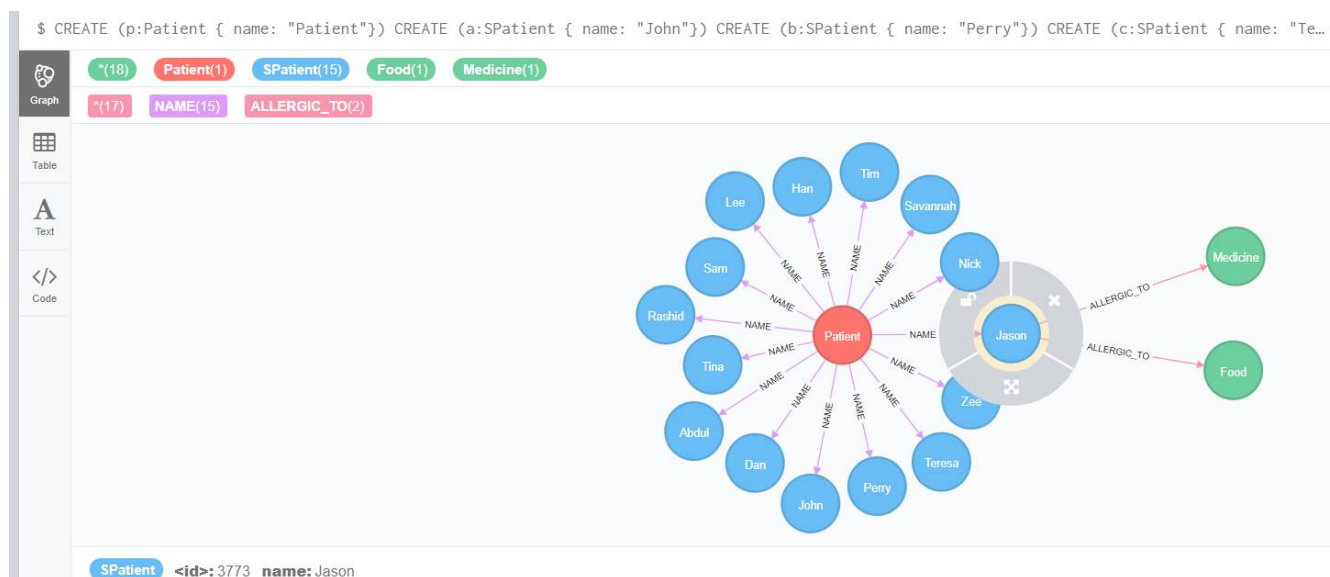


Fig 4.3 Extend patient's name node

Similarly, after extending food or medicine we see the exact component that person is allergic to using the relationship Types

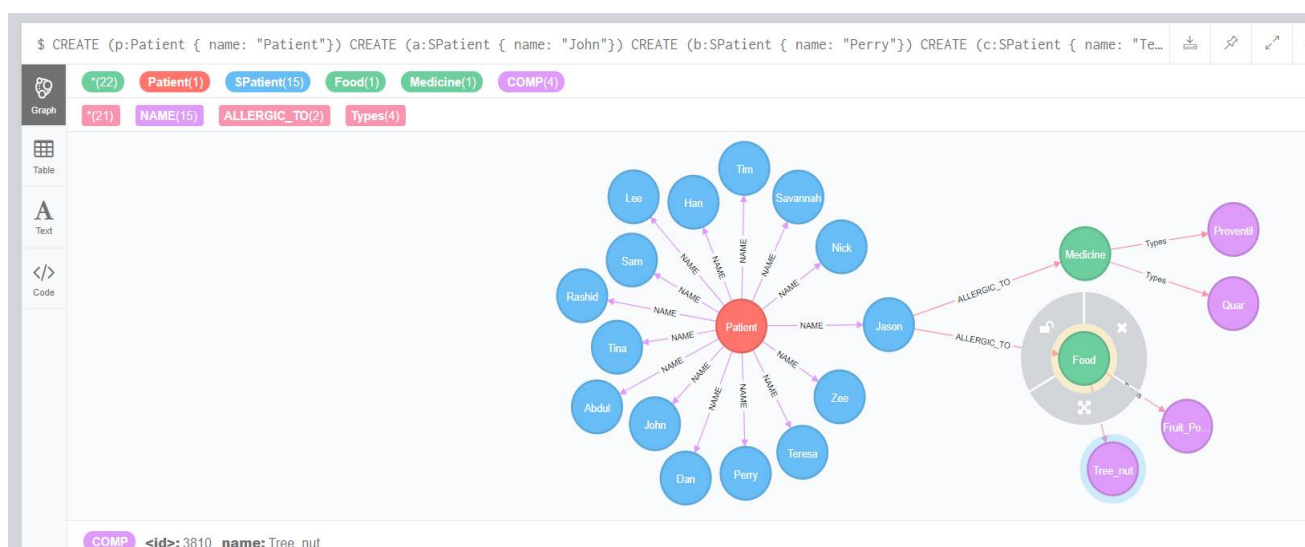


Fig 4.4 Extend Food and Medicine node

Thus, by using the extend feature, we can see the food and medicine components a particular patient is allergic to. If the number of patients in the parent node patient are looking too close to each other and it does not look proper, then another parent node named patient 1 can be created and some patients and their connected components are transferred to that parent node. We can also search a patient and find information directly related to that patient instead of finding a patient's name in all the patients.



ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(A High Impact Factor, Monthly, Peer Reviewed Journal)

Website: www.ijircce.com

Vol. 5, Issue 10, October 2017

V. CONCLUSION

Thus, we have implemented description of food and medicine components using NEO4J graphical representation by using the Cypher language. It is helpful for doctors and patients for proscribing the medicines which the patient is allergic to. Thus, the doctor can look at the graphical representation of database and decide what medicine and food components are to be given or not.

REFERENCES

- [1] <https://neo4j.com/product/>
- [2] <https://neo4j.com/product/#basics>
- [3] <https://neo4j.com/use-cases/fraud-detection/>
- [4] <https://neo4j.com/use-cases/identity-and-access-management/>
- [5] <https://neo4j.com/use-cases/knowledge-graph/>