Research Paper OnComplexity Reduction In Relational Database Using Neo4j Graph Database

T.P.Deshmukh[1], B.R.Bamnote[3], S.W.Ahmad[3]

[1]*(Computer Science And Engineering, Prof.RamMeghe Institute Of Technology And Research/ SantGadge Baba Amravati University, Badnera Maharashtra)*

[2]*(Computer Science And Engineering, Prof.RamMeghe Institute Of Technology And Research/ SantGadge Baba Amravati University, Badnera Maharashtra)*

[3]*(Computer Science And Engineering, Prof.RamMeghe Institute Of Technology And Research/ SantGadge Baba Amravati University, Badnera Maharashtra)*

Date of Submission: 05-12-2020 Date of Acceptance: 20-12-2020

ABSTRACT: Now a days, data is increasing in the volume because of digital process .more online transaction and new social network. There is large amount of data generated everyday in the field of manufacturing, business, science and our personal lives. Proper processing of the data could reveal new knowledge about market, society and environment, and enable us to react to emerging opportunities and changes in a timely manner. ,The ratio of increasing data amount in our digital world seems to outspeed the advance of our computing infrastructure. Databases are an indispenable part of a computing system and users heavily depend on the services they provide. When coact with a computing system, we expect that data be stored for future use, that the data is able to be looked up fastly, and we can perform complex queries and get the required data easily which is stored in the database. Many different types database types available for use such as relational databases, object databases, keyvalue databases, graph databases, and RDF databases. Each type of database has it's unique qualities that have applications in certain domains. It'swork aims to investigate and compare the performance and scalability of relational databases to graph databases in terms of handling multilevel queries such as finding the impact of a particular disease with the working area of pass out students. MySQL was selected as the relational database, Neo4j as the graph database.

KEYWORDS –Cypher Query Language, Graph DataBase, Relational Database, NoSql, Neo4j

I. INTRODUCTION

Since information technology is innovating on the way people live, the collection of digital data has started to grow rapidly. Today, there is tremendous amount of data generated everyday in the sectors of manufacturing, business, science and the personal lives. Proper processing of the data could reveal new knowledge about market, society and environment, and enable to react to emerging opportunities and changes in a timely manner. However, the growth of the data volume in digital world seems to outspeed the advance of computing infrastructure.

Today, the user data is increasing rapidly due to many data generating processes like latest social media networks, rapid adaptation of smartphones and handheld devices further enhances the data creation. The computation of this data is becoming more difficult day by day, as the users of the digital data and networks are increasing by manifolds. Traditional databases cannot compute this huge data without complexity for the real-time responses; whereas, in the case of graph databases, a graph is generated for each entity, which speeds up the process.

The relational model has dominated the computer industry since the 1980s mainly for storing and retrieving data.Of late, relational database has been losing its significance because of its dependence on a severe mapping which makes it hard to include new relationship between the entity. Another significant explanation of its disappointment is that as the accessible information is developing manifolds, it is getting confounded to work with relational model as joining an enormous number of tables isn't working effectively in relational databases, references to other rows and tables are indicated by referring to primary key attributes via foreign key columns. Joins are computed at query time by matching primary and foreign keys of all rows in the connected tables.

These operations are compute-heavy and memory-intensive and have an exponential cost..Today's chief information officer and chief technical officer don’t just need to manage larger volumes of data – they need to generate insight from their existing data.For this situation, the affiliation between data focuses matter more than the individual focuses themselves.

Relational databases, for example, Oracle and MySql dominate with regards to catching monotonous, even datain the tabular form. Despite the word “relational” in their name, relational database are much less effective at storing or expressing relationships between stored data elements. In contrast to a relational database, a graph database is organized altogether around data relationships. Graph databases treat relationship not as a schema structure yet as information, as different qualities.

In order to handle the issues of storing huge data, many of the researchers have presented the concept of graph and graph storage, in which the graphs are implied to model the huge data with complicated design. In every graph, there will be nodes, properties, and edges as the relationship among them. The connected data graph database also offers the significant choice to deal with the structured, semi-structure and unstructured data .

The graph database offers the fastest response to a query, many times, in milliseconds. Today, the graph databases are widely used in retail, social network, healthcare, communication and other online solutions. Operations like create, update, read and delete are available in graph database system. The drawback of these systems is that it is more expensive by nature than the traditional methods.

The graph database queries are domain-specific , user-friendly and can be treated as "SQL for graphs". The similarity to SQL is intentional and makes transition much easier for developers. When SQL execute query on the RDBMS is as long as half, the Cypher Query equivalent is much shorter. The traverser Application Programming Interfacce in RDBMS is highly resource intensive, since each step to neighboring node has to be depicted with JOIN. In the graph database hypergraph property allows direct access to adjacent nodes by eliminating the edge attribute.

There are several graph databases like Neo4j Graph Database, OrientDB ,ArangoDB , MarkLogic and Allegro Graph. Neo4j is most popular and open source graph database.Its native and elegant solution and ease of data remodeling.

Notwithstanding of relational database, and well known graph database, Neo4j utilizes graph related concept to portray data models,the data can undoubtedly be nodes, edges and their related attributes.. By focusing on medical data. Firstly, Neo4j and Cypher Query Language are introduced. Then Neo4j is applied to analyze the associations among key objects in Hospital data which are Doctor, Patient , diseases etc. Neo4j database is good at dealing with complex and multi-connection data, using Neo4j database to store and manage medical data makes it convenient for Hospital data analysis.

1.1 MOTIVATION:

Databases are an integral part of a computing system and users heavily rely on the services they provide. When interact with a computing system, It’s expect that data be stored for future use, that the data is able to be looked up fastly, and . Now a days data is increasing in volume as new digital processes, morewe can perform complex query against the data held in the database. A wide range of rising database types accessible for utilize, for example, social relational databases, object database, key-value database, chart graph databases, and RDF databases. Each sort of database gives extraordinary characteristics that have applications in specific domains.

To exlore the concept of Neo4j Graph Database proposed the system to develope the application for Hospital Information System. Hospital data analysis is important for application to find relationship between health data.

Work aims to examine the performance of relational databases to graph databases in terms of handling multilevel queries online transaction ,new social network and more devices are introduced. And thus, customers ,products,processes, devices are getting more connected , interact and relate to each other.

1.2 OBJECTIVE:

Following are the proposed objective:

1. It enables to reduce the complexity of relational database.
2. DBA will be able to monitor the operation with ease.
3. It delivers the lighting fast read and write performance needed by application, while still protecting data integrity.
4. Capable of handle a highly a connected data and increase in the volume.
5. Obtain minutes to milliseconds performance for data retrival.

II. LITERATURE REVIEW

Concerning the origins of graph databases, K. Nagi. [2]developed a review on graph database models proposed before the year 2002. The authors synthesized the notion of a "graph database model" and compare the proposals available at the moment. It is important to emphasize that most of the works reviewed by the authors followed a theoretical interest more than practical developments. Hence, a practical evaluation of the models is not available. With respect to the recent developments in the area.

Pere Burton [10] reviewed six graph databases (Neo4j, HyperGraphDB, DEX, InfoGrid, Sones and VertexDB) and published a comparison-matrix that included information like software features (e.g., license), schema features (e.g., types of nodes and edges), query features (e.g., language and traversals), general database features (e.g., transactions, indexing), database operation utilities (e.g., protocols), language bindings and operating systems. This work summarizes the features but does not include major discussion nor analysis. Another informal review Pere Burton included an interesting questionnaire about desirable features for graph databases.

J.Webber[3] evaluated the performance of three graph databases (Neo4j, HypergraphDB and DEX) and an RDF Database (Jena). The tests, that included the evaluation of several typical graph operations over different graph sizes, shown that DEX and Neo4j were the most efficient implementations.

The thought learning system considering base exposure is Graph Based Induction (GBI) (Yoshida, Motoda, 1995). It employs coloured digraph as the representation framework where colours attached to the nodes represent the attributes of the facts. GBI analyzes each associated pair of nodes, and consolidations the continuous typical ones. The final merged substructures are labelled as concepts. He evaluated the execution of graph databases (Neo4j, HypergraphDB and DEX) and a RDF Database (Jena). The tests, that incorporated the assessment of a few average graph operations over various graph sizes, demonstrated that DEX and Neo4j were the most efficient implementations.

B.T. Messmer et al. presents a decision tree approach for indexing models for isomorphism and subgraph Isomorphism. This strategy creates replies in polynomial time, at the cost of a record which is exponential in measure as for database size.

Angles, R.; et al presents a survey of earlier work (pre-NOSQL) in graph databases. i.e. prior to 2002,particularly geographical, spatial and semi structured database models. More established dada models zeroed in vigorously on semi organized and XML data in a traditional database.The authors orchestrated the thought of a "graph database model" and analyze proposals accessible right now

. Angles, R.; performs comparison & performance analysis of different graph database models compares current graph databases concentrating on their data model features, that is data structures, query facilities, and integrity constraints. Author shows that most graph database models provide an innate support for different graph structures, query facilities in the form of APIs (most of the models) and query languages (a few of them), and basic notions of integrity constraints[6].

Philip Howard, analyst at Bloor Research definite that Graph databases are fundamental when the degree of division. Howard remarked on the constraints of graph databases, saying that "The real impediment is that while these are actually NoSQL databases, practically speaking they can't be executed over a negligible exertion gathering (at any rate not a present) but instead need to continue running on a lone machine, the explanation being that execution adulterates rapidly over a framework.Another potential drawback is that conceivably you have to make your own inquiries using Java or whatever — which suggests using expensive programming engineers — or you use SparcQL or one of the other request vernaculars that have been made to help chart information bases, anyway this infers taking in another skill.

Jouili, S.etal precisely contemplates graph databases ie shows Graph Database Benchmark, to see four graph databasesNeo4j, DEX, Titan (BerkeleyDB and Cassandra) and OrientDB (neighborhood) on various sorts of workloads, each time perceiving which datbase was the best and the less balanced. In light of measure, the database that got the best outcomes with traversal workloads is certainly Neo4j: it outflanks the various competitors, in any case the workload or the parameters used[6].

General systems for making designed records for benchmarking and testing record linkage approaches are depicted in (Christen and Vatsalan, 2013; Ioannou, Rassadko, and Velegrakis, 2013; Talburt, Zhou, and Shivaiah, 2009). The data generation tool used in this work is specialised for genealogical population structures, and the global configuration parameters that are supported reflect this.

Healthcare system in United States was generating more data and they required new technology to handle data analytics effectively. Data driven approach is used to handle data analytics in healthcare systems by using two independent tasks, data management and data services. Here,data management means storing the data with minimal redundant structure and error free. Data services portray different investigation inquiries, for example, join, search and measurable questions

The problem appeared due to the gap between data management and data services in relational databases. To overcome this problem, they presented an approach to convert third normal form (3NF) of relational databases in equivalent graph of Graph database. A graph database does not require creating more tables and replicating them unlike relational databases. For example, Neo4j is suitable in OLTP (online transaction processing) environment. Pregel is utilized where high latency and high through put have high need. The experiments have shown that Graph database performed better than relational database (MySQL) in the heterogeneous environment of healthcare systems of United States in OLTP

The impediments of conventional database, specifically the relational model, to cover the prerequisites of current application areas, has lead the improvement of new innovations called NOSQL database.. According to its data model, these databases can be categorized as: Widecolumn stores, which follow the BigTable model of Google (e.g., Cassandra); Document stores, which are oriented to store semi-structured data (e.g., MongoDB); Key-value stores, which implement a key to value persistent map for data indexing and retrieval e.g. BerkeleyDB&#41;; and Graph Databases, which are oriented to store graph-like data.

Activity around graph databases flourished in the first half of the nineties and then the topic almost disappeared [19]. As of late the territory is picking up consideration on the grounds that in popular activities where an database is required, the significance of the data depends on the relations more or equivalent than on the elements (an essential standard of each graph database). Also, the proceeded with development and increment of huge and complex graph like database makes a graph database a critical necessity. This renascence is appeared by the accessibility of a graph databases frameworks.

In 1964 , Charles Bachmann developed the first commercial database management system (DBMS) , IDS - Integrated Data Store based upon an early network data model while working at Honeywell [20]. IDS maintained a single set of shared files on disk. Programs responsible for particular tasks, such as billing or inventory updates, retrieved and updated these files by sending requests to IDS. IDS provided application programmers with a set of powerful commands to manipulate data. In the late 1960s, IBM and North American Aviation (later Rockwell International) developed the first commercial hierarchal DBMS, IMS - Information Management System, and its DL/1-language. These models lack a good abstraction level which implies that it is difficult to separate the db-model from the actual implementation. The data structures provided are not apt for modelling non traditional applications.

They permit database navigation at the record level by providing low-level operations that can be used to derive more abstract structures. The solution to these problems was given by Codd. In 1970, Codd released ―A Relational Model of Data for Large Shared Data Banks.‖ which provided the first definition of Relational Model. Codd's relational model suggested that all data in a database could be represented as a tabular structure (tables with columns and rows, which he called relations) and that these relations could be accessed using a high-level non-procedural (or declarative) language [reference].

Edgar F. Codd published a series of papers about relational databases from 1970 to 1972 [21,22,23] in which he focused on the concept of abstraction levels by introducing a separation between the physical and logical levels. It was a major development because it gave the data modelling discipline a mathematical foundation. It is based on the simple notion of relation, which together 14 with its logic and algebra, made it the most widely used model and its standard query language SQL, became a paradigmatic language for querying. But with time, database community realized that the relational model was inadequate for data intensive domains (knowledge bases, engineering applications) involving complex data objects and object interactions such as CAD/CAM software, computer graphics, and information retrieval. In order to provide additional semantics necessary to model these new applications Object-oriented (O-O) db-models appeared in the eighties .

As per Rao in 1994, "The Object Oriented database (OODB) worldview is the object oriented programming language (OOPL) frameworks and determined frameworks .

O-O dbmodels view the world as a set of complex objects having certain states (data), where interaction is via method passing.In spite of the fact that O-O db-models grant a lot more extravagant structures than the relational db-model, they actually necessitate that all information adjust to a predefined schema. One of the promising solutions to this problem was given by document databases such as Lucene, which is able to fully index large document collections and support queries that rank the documents according to information retrieval measures. But what makes document databases really different, is the fact that documents are usually retrieved through dynamic and unpredictable queries. Thus document databases can usually associate any number of fields of any length to a document. O-O dbmodels view the world as a set of complex objects having certain states (data), where interaction is via method passing.

In spite of the fact that O-O db-models grant a lot more extravagant structures than the relational db-model, they actually necessitate that all information adjust to a predefined schema. One of the promising solutions to this problem was given by document databases such as Lucene, which is able to fully index large document collections and support queries that rank the documents according to information retrieval measures. But what makes document databases really different, is the fact that documents are usually retrieved through dynamic and unpredictable queries. Thus document databases can usually associate any number of fields of any length to a document.

Shipman in 1981 presented an implicit structure of graphs for the data itself in the Functional Data Model, whose goal was to provide a ―conceptually natural‖ database interface. Logical Data Model (LDM) developed by Kuper and Vardi in 1984 [24], was proposed on a different approach according to which an explicit graph db-model intended to generalize the relational, hierarchical and network models. Later Kunii in 1987 [25] proposed a graph dbmodel for representing complex structures of knowledge called G-Base.

In the late eighties, Lecluse in 1988 introduced O2 [26], an object-oriented db-model based on a graph structure. Along the same lines, GOOD developed by Gyssens in 1990 was a system in which manipulation as well as representation are transparently graph based. Among the subsequent developments based on GOOD are: GMOD by Andries in 1992, which proposes a number of concepts for graph-oriented database user interfaces;

Gram by Amann and Scholl in 1992, which is an explicit graph dbmodel for hypertext data [27]; PaMaL by Gemis and Paredaens in 1993, which extends GOOD with explicit representation of tuples and sets; GOAL developed by Hidders and Paredaens in 1993, which introduces the notion of association nodes; GLog in 1995, proposed a declarative query language for graphs; and GDM developed by Hidders in 2005 which incorporates representation of n-ary symmetric relationships .

B. A. Sutrisna, W. K. RahmatSaleh and A. A. Gozali, "Implementation of GRAC algorithm (Graph Algorithm Clustering) in graph database compression," Information and Communication Technology (ICoICT), 2015 3rd International Conference on, Nusa Dua, 2015, pp. 391-395. States that graph database offers the fastest response to a query, many times, in milliseconds. Today, the graph databases are widely used in retail, social network, healthcare, communication and other online solutions. Operations like create, update, read and delete are available in graph database system. The drawback of these systems is that it is more expensive by nature than the traditional methods.

Neo4j was fully written in Java and can be deployed on multiple systems . 17 Neo4j's founder Emil Eifrem claims that socially enabled applications are gravitating towards graph databases because other types of databases are not effective for managing relationships between millions of users with multiple connections. Neo4j is the ideal solution for any application that relies on the relationships between records. Many graph databases are huge and rising rapidly in size. For example, the social networking site Facebook contains a large network of registered users and their friendships. The number of Facebook users has grown from less than 5 million in September 2005 to close to 10 million in September 2006, then to 50 million in September 2007.

There is a critical need for efficient and effective graph querying systems to query and mine these growing graph databases. Although the effective usage of Graph databases has been growing manifold, the problem faced is in querying the graph databases efficiently and effectively. After going through various proposals presented for graph databases, it has been realized that out of all the databases available, Neo4j is apt for representing and retrieving data for the World Wide Web. This thesis evaluates the performance of Graph databases in comparison to relational databases by designing graph databases in Neo4j. Here, a comparison of query retrieval times of relational and Neo4j database has been conducted. The evaluation measures not only include the query retrieval times but scalability, flexibility and maturity and accuracy in search results as well in minimum time.

"A hospital resource and patient management system based on real-time data capture and intelligent decision making" Author(s): Musa, A. Lancashire Bus. Sch., Univ. of Central Lancashire, Preston, UK Yusuf, Y, Meckel.M. Systems and Informatics (ICSAI), 2012 International Conference.This paper highlights such limitations of existing systems and proposes a RFID(Radio Frequency ID) and wireless sensor based , location and information management framework that facilitates real time tracking of hospital assets, personnel and patients as they move through pre-set procedures as part of daily activities of the hospitals.

"Study on information system of health care services management in hospital" Author(s): Daiping Hu, Antai Sch. of Manage., Shanghai Jiaotong Univ., China Weiguo Xu ; Huizhang Shen ; Mengyu Li. Services Systems and Services Management, 2005. Proceedings of ICSSSM '05. 2005 International Conference.This paper reviews the HIS (Hospital Information Systems) which are widely used in many hospitals in China mainly to provide easier and faster way for daily medical tasks /activities with a GUI And provides for overcoming some of the limitations of HIS , eg. HIS aims at improving quality of health care services but do not have way of evaluating /measuring those.

"Specification of a Reference Model for the Domain Layer of a Hospital Information System"Author(s): Gudrun Hübner-Blodera , ElskeAmmenwertha , Birgit Brigl b , Alfred Winter b a Institute for Health Information Systems,UMIT – University for Health Sciences, Medical Informatics and Technology, Hall in Tyrol, Austria b Institute for Medical Informatics, Statistics and Epidemiology, University of Leipzig, Germany, ENMI, 2005.This paper aims at creating a reference data model that will serve as a generic starting point for any new HIS development projects so costs involved in studying and analyzing current state and coming up with gaps analysis and additional requirements can be significantly reduced.

"Developing Effective Hospital Management Data Systems: A Technology Ecosystem Perspective". DATE OF SUBMISSION: 5 October 2014 This paper more so contributes to Planning, Design and development aspects of any Hospital management system by highlighting ESFs that should be considered.

Hospital management system consist of different softwares that are integrated in order to capture data in specific sections of the hospital [Garrido, Raymond, Jamieson, Liang and Wiesenthal [2004:21-22]],It is supposed to make the right information and knowledge available to the right people, in the right place, at the right time and in the right form.

III. SYSTEM ANALYSIS:

Hardware And Software Specification:

Hardware Specification:

Processor: i7 core
RAM: 4GB
Hard Disk: 100GB

Software Requirement:

Operating System: Windows 7
Server Side Script: php
Front End: Html 5, jquery
Back End : My sql, Neo4j database

IV. SYSTEM DESIGN:

4.1 Proposed System Architecture:

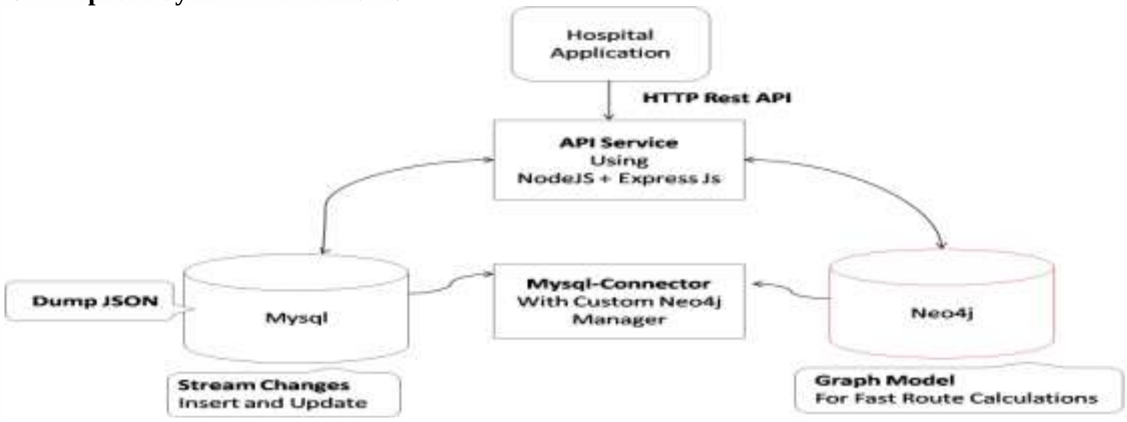

Hospital Application
HTTP Rest API
API Service Using NodeJS + Express Js
Dump JSON
Mysql
Mysql-Connector With Custom Neo4j Manager
Neo4j
Stream Changes Insert and Update
Graph Model For Fast Route Calculations

Fig. 4.1 Proposed System Architecture

Above figure shows the proposed system architecture. For the hospital application we are using Neo4j graph database for manipulation and retrival. Graph databases can serve as an alternate for relational databases for determining relationships optimally and quickly in the hospital application system.

Neo4j graph database is used for the manipulation, retrieval of data. It's a NoSQL database that belongs to the category of graph databases and follows a structure where the nodes are represented as vertices and the relationships as edges. It model and store the data as graph and seamlessly evolve applications to support new requirement by adding new kinds of data and relationship. Cypher is the Neo4j's graph query language that allows users to store and retrieve a data from graph database.

As Neo4j has a flexible structure defined by stored relationships between data records .With Neo4j, each data record, or node, stores direct pointers to all the nodes it's connected to.Since Neo4j is planned around this straightforward, yet ground-breaking improvement, it performs queries with complex associations significant degrees quicker, and with more profundity, than different databases.

In relational database like SQL we need different tables to represent the hospital localization application. In the event that we use Neo4j for the medical clinic restriction application the clinics can be represented as nodes and the association between them by street can be represented to as the connection between them. We can have specializations in a hospital as properties for the nodes and the distance between the hospitals as the property for the relationship.

With Neo4j, associations between data are stored – not processed at the moment of query time. Cypher is an amazing, graph enhanced query language that comprehends, and exploits, these put away associations. When attempting to discover examples or bits of patterns inside information, Cypher quries are regularly a lot more straightforward and simpler to compose than enormous SQL JOINs. Since Neo4j doesn't have tables, there are no JOINs to stress over. For examination with SQL, here's a simple Cypher query matching all products in a category hierarchy:

In Neo4j we use cypher query language to build up the hospital information system as graph database . It follows SQL like syntax, which is basic and comprehensible. Let's now see the implementation of Neo4j. The procedure for implementing hospital information applicaiton using Neo4j is given below:

1. Create nodes representing entity of the hospital information system
2. Associate each entity with the specializations as properties
3. Associate nodes with relationship and add distance as property to the relationship

4.2 Proposed System Design

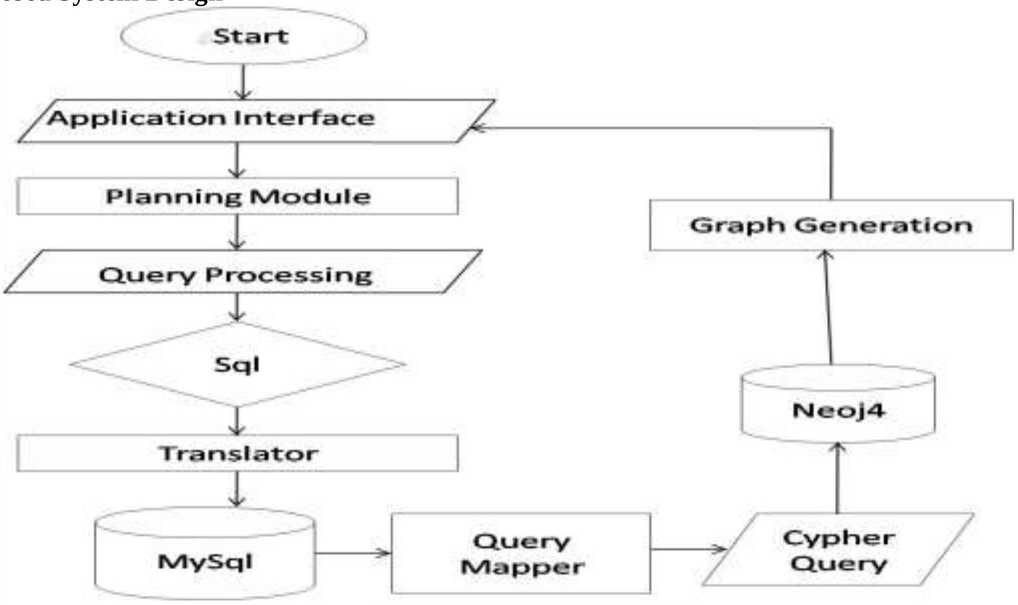


Fig. 4.2 Proposed System Design

Above figure shows the proposed system design.

An application interface is a computing interface which defines interactions between multiple software intermediaries like between Mysql and Neo4j graph database. It characterizes the sorts of solicitations that can be made, the information organizes that ought to be utilized, the shows to follow, and so forth It can likewise give expansion instruments so clients can broaden existing usefulness in different manners and to changing degrees.

An API can be completely custom, explicit to a part, or it very well may be planned dependent on an industry-standard to guarantee interoperability. Through data stowing away, APIs empower secluded programming, which permits clients to utilize the interface freely of the usage.

Hospital information system includes some of the modules which is as follows:

- Doctor login and registration
- Patient login and registration
- Medical care
- MCI
- Services Module

Query preparing signifies the arrangement and execution of an query particular normally communicated in a decisive database query language, for example, the structured question language (SQL). Query handlingcomprises of an incorporate time stage and a runtime stage. At compile time, the compiler makes an interpretation of the query particular into an executable program. This interpretation cycle (regularly called inquiry aggregation) is contained lexical, linguistic, and semantical examination of the query determination just as a question improvement and code age stage.

These administrators actualize information access, joins, choices, projections, gathering, and accumulation. At runtime, the database engine interprets and executes the program implementing the query specification to produce the query result The code generated usually consists of physical operators for a database machine.

Collected data of the hospital information system will be stored in the Mysql database. SQL isa domain-specific language used in programming and designed for managing data held in a relational database management system i.e Mysql.

Csv is a comma-seperated files which is delimited content document that utilizes a comma to isolate esteems. Each line of the document is an information record. Each record comprises of at least one fields, isolated by commas. The utilization of the comma as a field separator is the wellspring of the name for this record format.
Cypher is Neo4j's graph query language that allows users to store and retrieve data from the graph database. Neo4j will provide the result in the form of graph.

4.3 Proposed Algorithm:

Step 1: Start
Step 2: Develop Planning Module

Step 3: Data Store In SQL
Step 4: Import Data Into Neo4j Database From Mysql Database
Step 5: Neo4j Database Will Generate The Data In Grphical Format.
Step 6: Stop

In hospital information system , there are 4 modules as follows.

Doctor login and registration : - This Registration module is an integrated doctor management system, which captures complete and relevant doctors information. The system automates the doctors administration functions to have better and efficient service process. With this module doctor can register their name and login as well.

MCI : - MCI is a Medical Council of India which is a statutory body for establishing uniform and high standards of medical education and recognization of medical qualifications in India.

Patient login and registration : - This Registration module is an integrated patients management system, which captures complete and relevant patients information. The system automates the pateints administration functions to have better and efficient service process. With this module patient can register their name and login as well.

Medical care login :- This module is an integrated medicine, pharmacy management system, which captures complete and relevant patients and his treatment information. The system automates the administration functions to have better and efficient service process. With this module pharmacist can register their name and login as well.

We will stored collected data of hospital information system into Mysql database. We can import data into neo4j by two ways. We have imported data into neo4j by using Load csv command. In the query console panel of neo4j we can write queries.

Neo4j will provide the result in graphical format with lightning-fast read and write performance needed while still protecting the data integrity for the given queries.

V. SYSTEM SPECIFICATION:

5.2.1 Input Specification: The system is classified into subsystem such as:

- Doctor login and registration
- Patient login and registration
- Medical care registration
- MCI page

5.2.2 Output Specification:

Neo4j graph database will generate the data in the form of graph which is stored in relational database.Itdelivers the lightning-fast read and write performance needed while still protecting the data integrity. It compare the performance and scalability of relational database to graph database.

VI. SYSTEM IMPLEMENTATION DETAILS:

6.1 Neo4j graph database creation details

Following figure shows the Neo4j browser before graph creation. It shows 0 nodes(0 labels) and 0 relationship(0 types).

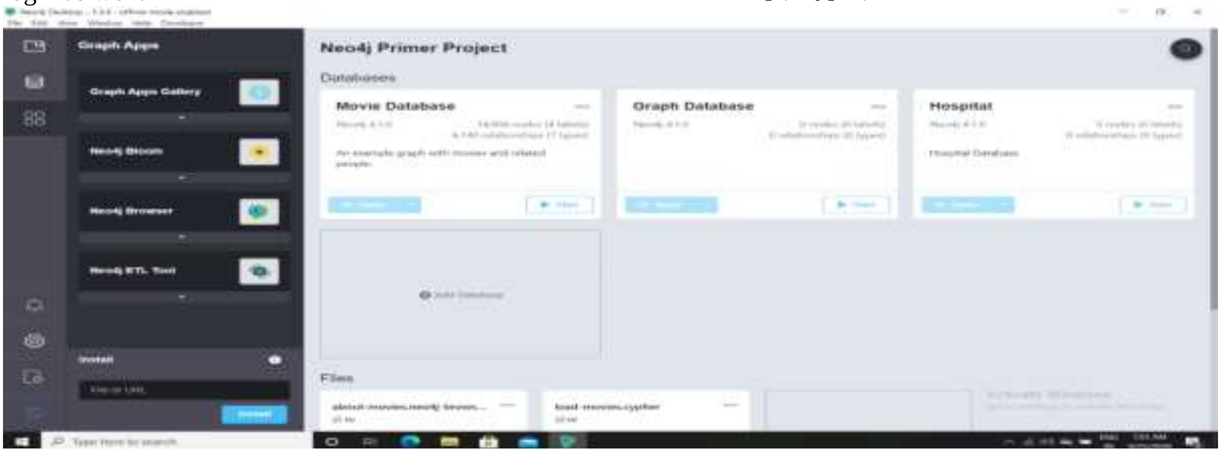
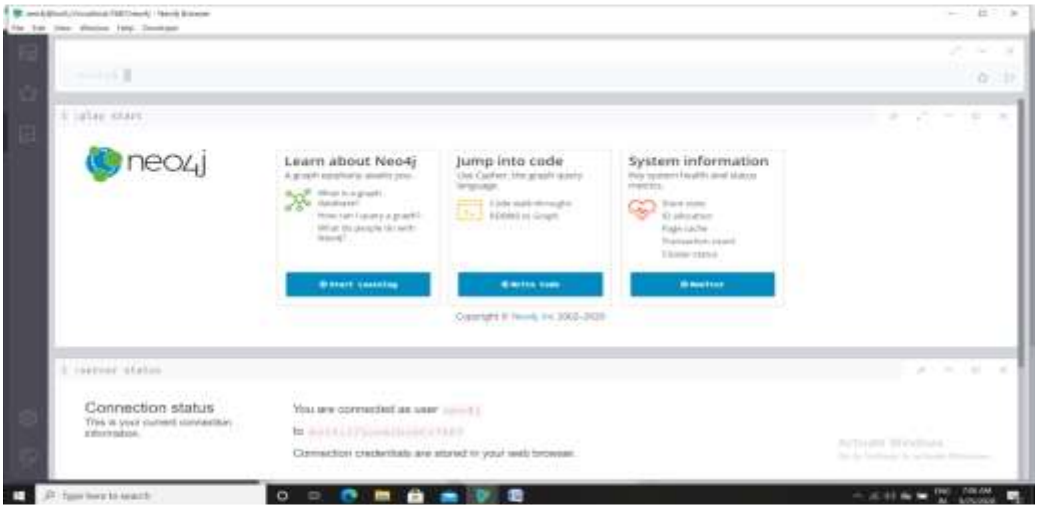


Fig.6.1 Neo4j Browser before Graph Creation

Fig 6.2 showsQuery Console. Neo4j console is a web – based graph that acts as an interactive panel, where we can import data and modify it by entering statements in the input field at the bottom in query console panel.following figure shows the query console.

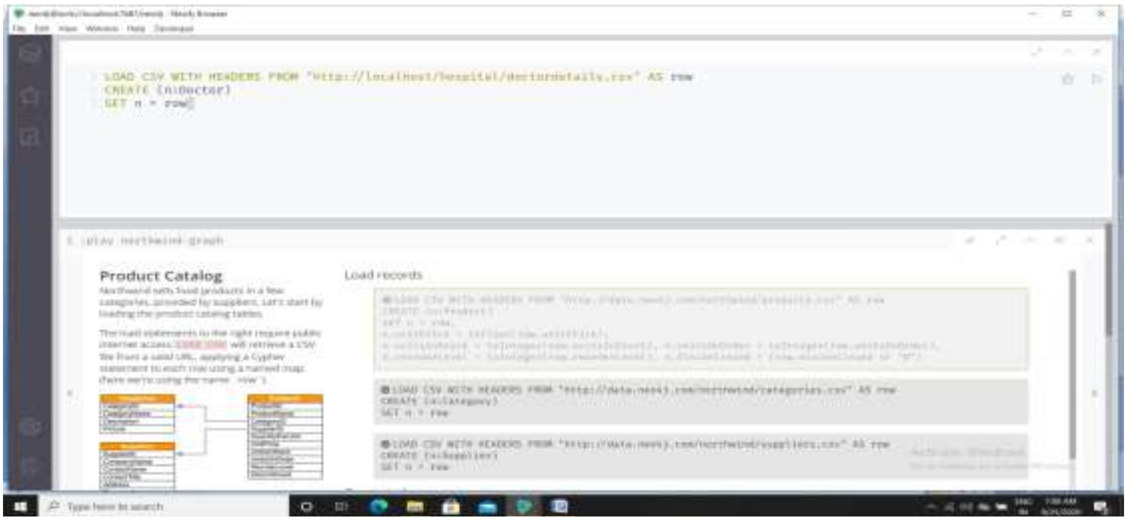


You are connected as user

Connection credentials are stored in your web browser.

Fig 6.2 Query Console

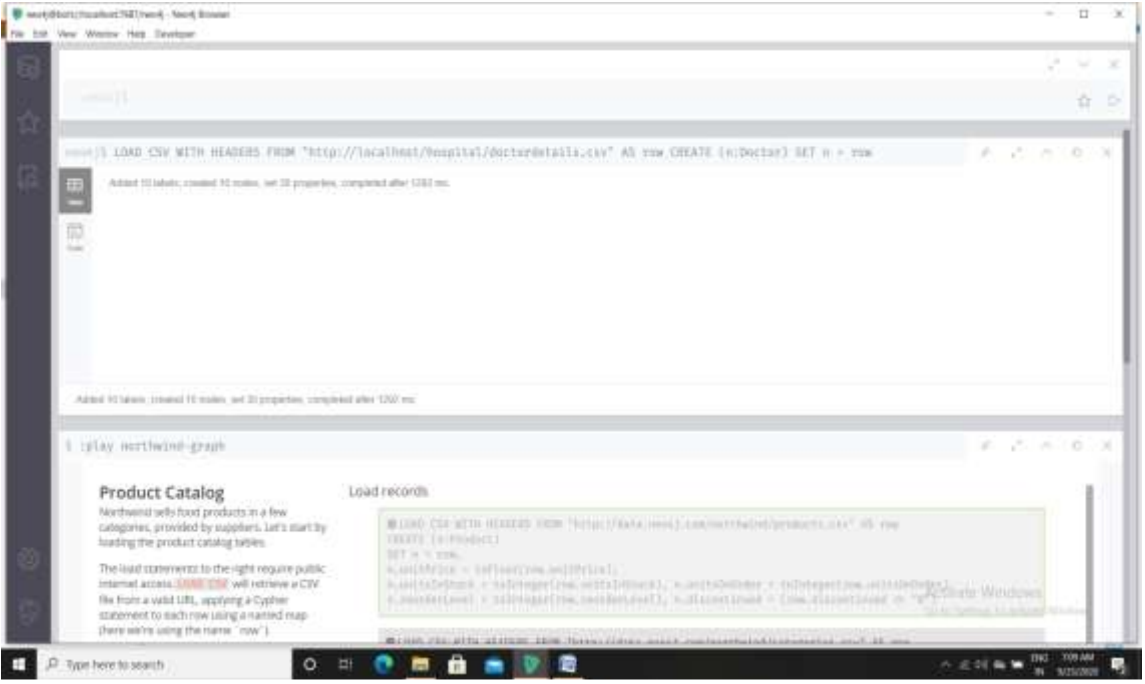
Figure 6.3 shows query for loading the CSV file which is named as Doctor Details. CSV is a comma-separated values file which allows data to be saved in a tabular format often viewed in Excel or some other spreadsheet tool.



```
LOAD CSV WITH HEADERS FROM "http://localhost/hospital/doctordetails.csv" AS row
CREATE (n:Doctor)
SET n = row
```

Fig 6.3 Query for Loading the CSV file Doctor Details

Figure 6.4 Execution of CSV file shows the execution of CSV file.



```
LOAD CSV WITH HEADERS FROM "http://localhost/hospital/doctordetails.csv" AS row CREATE (n:Doctor) SET n = row
```

Added 10 labels, created 10 nodes, set 20 properties, completed after 1202 ms.

```
:play northwind-graph
```

Product Catalog

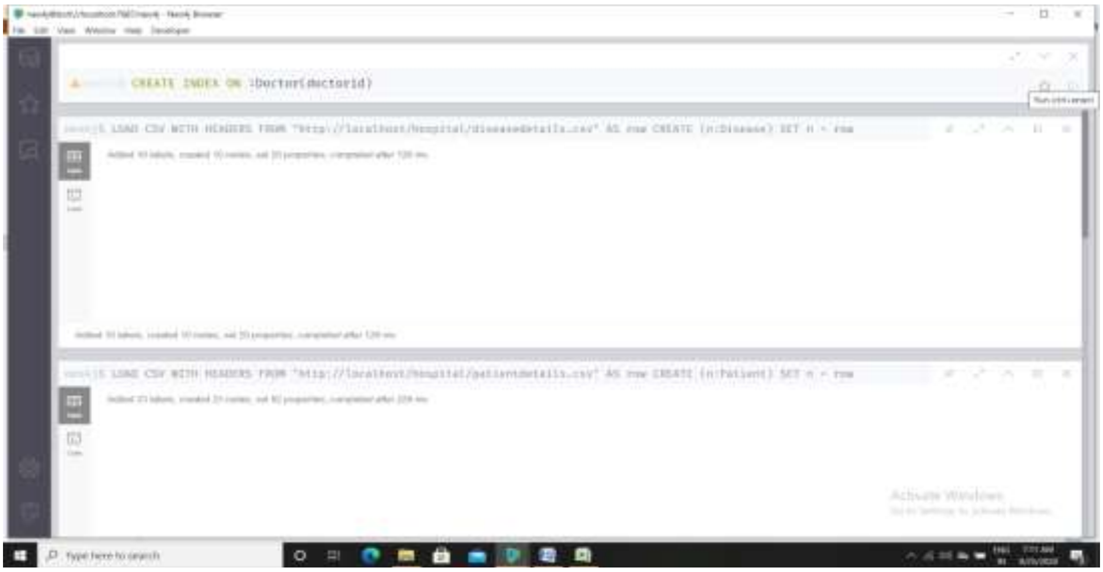
Northwind sells food products in a few categories, provided by suppliers. Let's start by loading the product catalog tables.

The load statements to the right require public internet access. LOAD CSV will retrieve a CSV file from a valid URL, applying a Cypher statement to each row using a named map (here we're using the name `row`).

Load records

```
LOAD CSV WITH HEADERS FROM "http://data.neo4j.com/northwind/products.csv" AS row
CREATE (n:Product)
SET n = row,
n.unitPrice = toFloat(row.unitPrice),
n.unitsInStock = toInteger(row.unitsInStock), n.unitsOnOrder = toInteger(row.unitsOnOrder),
n.reorderLevel = toInteger(row.reorderLevel), n.discontinued = (row.discontinued <> "0")
```

Fig. 6.4 Execution of CSV file



```
CREATE INDEX ON :Doctor(doctorid)
```

```
LOAD CSV WITH HEADERS FROM "http://localhost/hospital/diseasedetails.csv" AS row CREATE (n:Disease) SET n = row
```

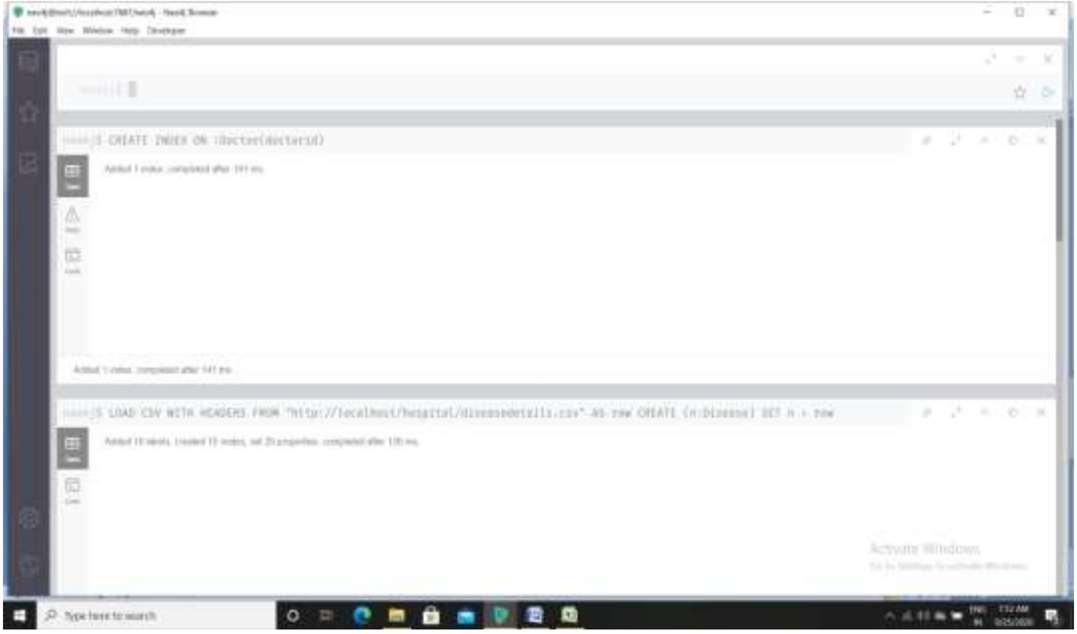
Added 10 labels, created 10 nodes, set 20 properties, completed after 120 ms.

```
LOAD CSV WITH HEADERS FROM "http://localhost/hospital/patientdetails.csv" AS row CREATE (n:Patient) SET n = row
```

Added 21 labels, created 21 nodes, set 82 properties, completed after 228 ms.

Fig. 6.5 Create Indexing on CSV file

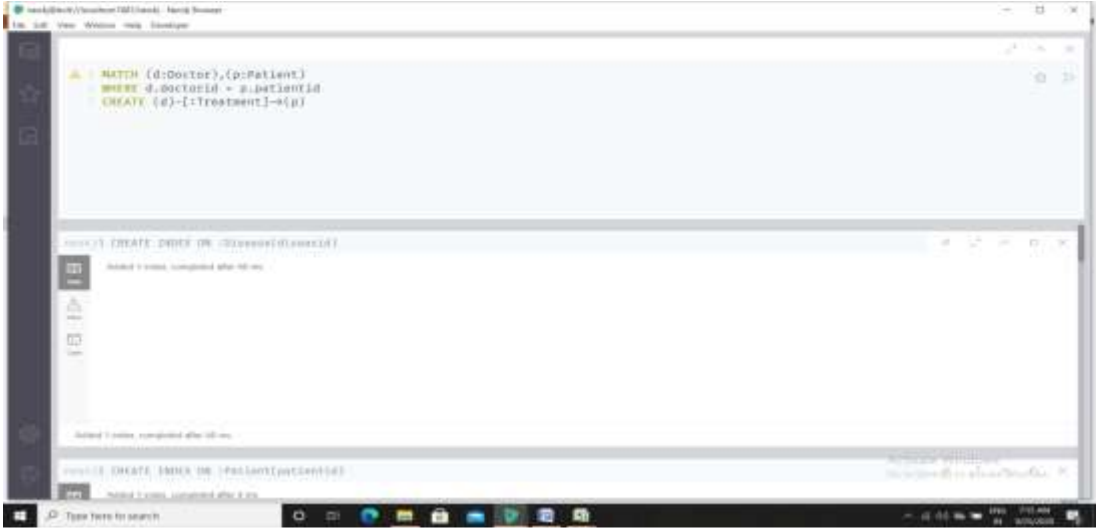
Figure 6.6 shows the execution of indexing on CSV file



```
$ CREATE INDEX ON :Doctor(doctorid)
Added 1 index, completed after 141 ms.

Added 1 index, completed after 141 ms.

$ LOAD CSV WITH HEADERS FROM "http://localhost/hospital/diseasedetails.csv" AS row CREATE (n:Disease) SET n = row
Added 11 labels, created 11 nodes, set 21 properties, completed after 120 ms.
```

Fig. 6.6 Execution of Indexing on CSV file

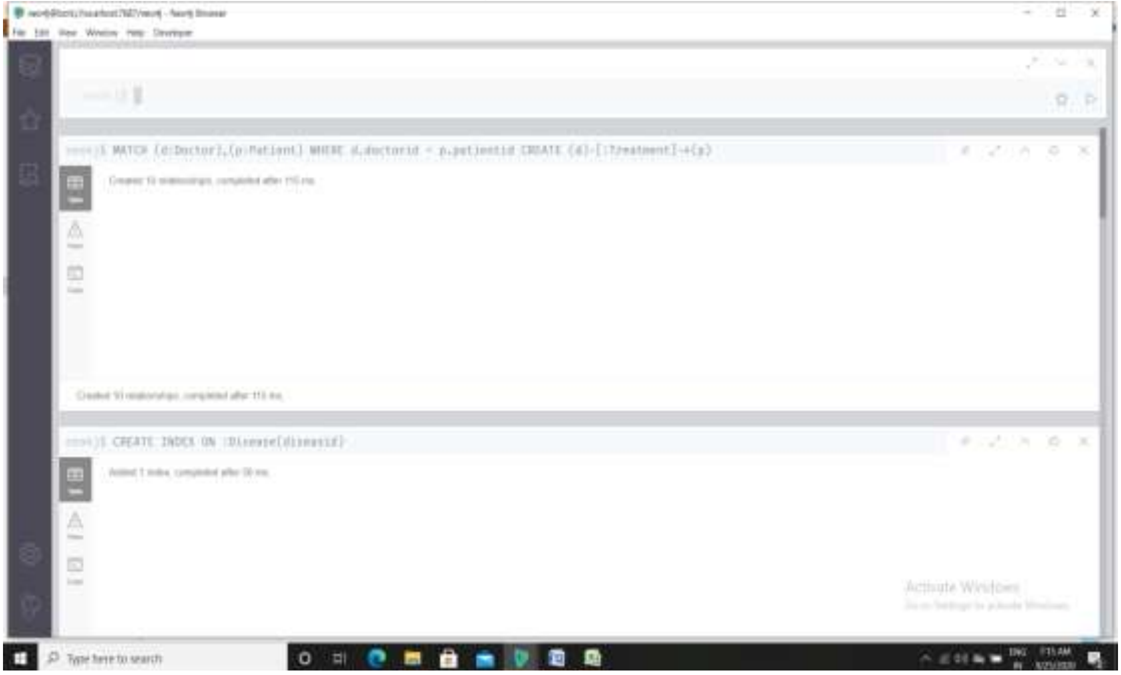
Following figure shows theMatching of Node with different CSV Files



```
MATCH (d:Doctor),(p:Patient)
WHERE d.doctorid = p.patientid
CREATE (d)-[:Treatment]->(p)

$ CREATE INDEX ON :Disease(diseaseid)
Added 1 index, completed after 48 ms.

Added 1 index, completed after 48 ms.

$ CREATE INDEX ON :Patient(patientid)
Added 1 index, completed after 8 ms.
```

Fig. 6.7 Matching of Node with different CSV Files

Following figure shows the execution of matching node.



```
MATCH (d:Doctor),(p:Patient) WHERE d.doctorid = p.patientid CREATE (d)-[:Treatment]->(p)
```

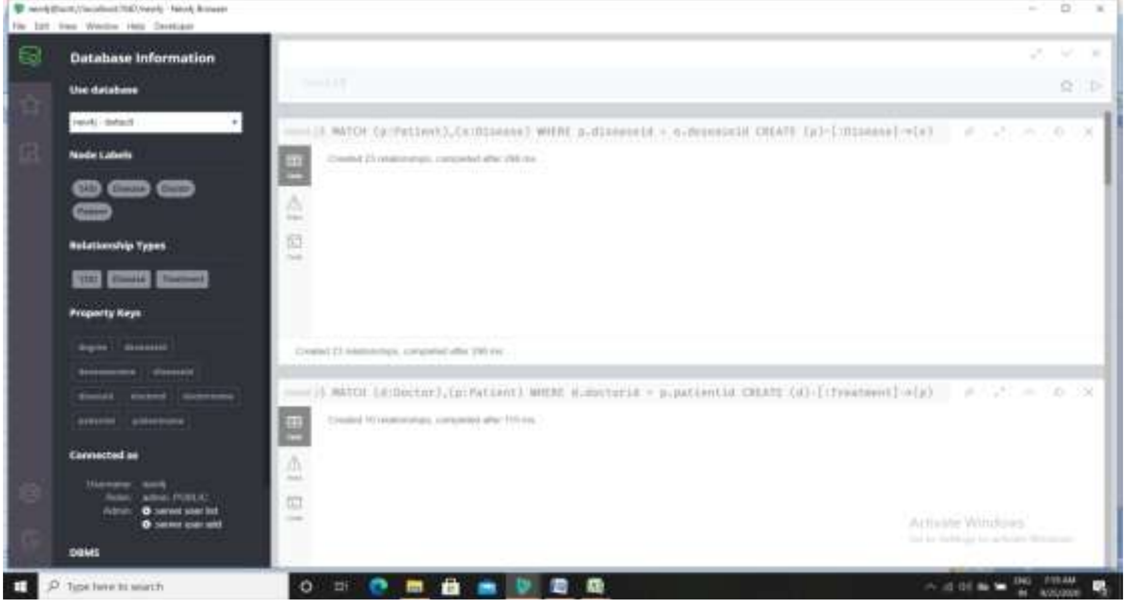
Created 10 relationships, completed after 115 ms.

```
CREATE INDEX ON :Disease(diseaseid)
```

Added 1 index, completed after 30 ms.

Fig. 6.8 Execution of Matching Node

Figure 6.10shows graph execution panel.



Database Information

Use database: neo4j - default

Node Labels

Relationship Types

Property Keys

Connected as

Username: neo4j
Roles: admin, PUBLIC
Admin: :server user list
:server user add

DBMS

```
MATCH (p:Patient),(s:Disease) WHERE p.diseaseid = s.diseaseid CREATE (p)-[:Disease]->(s)
```

Created 23 relationships, completed after 290 ms.

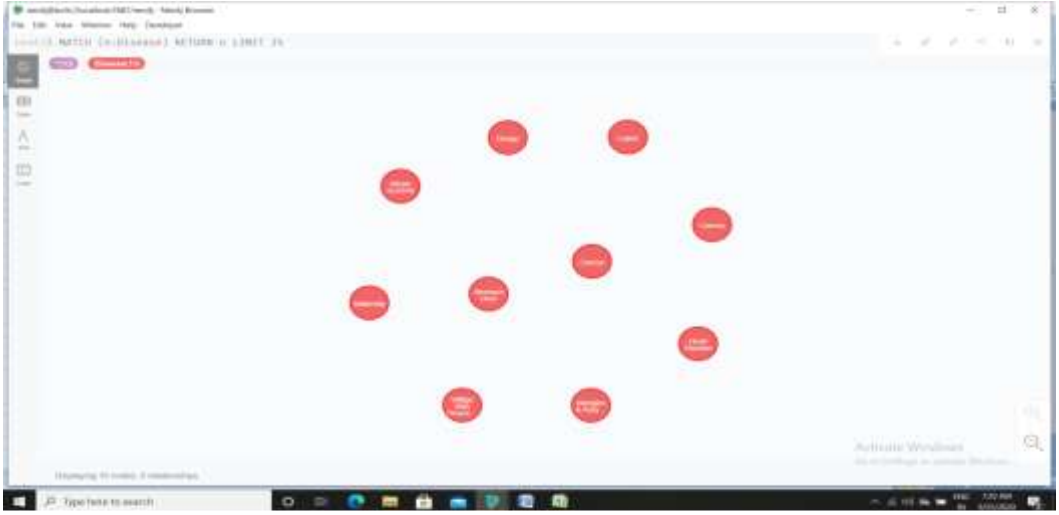
```
MATCH (d:Doctor),(p:Patient) WHERE d.doctorid = p.patientid CREATE (d)-[:Treatment]->(p)
```

Created 10 relationships, completed after 115 ms.

Fig. 6.10 Graph Execution Panel

Neo4j Bloom is a data exploration tool that visualizes data in the graph and allows users to navigate and query the data without any query language or programming. Users can write patterns similar to natural language questions to retrieve data and traverse layers of the graph. Bloom also allows appropriate users to edit, update, or correct the graph when missing informtion or bad data is found.

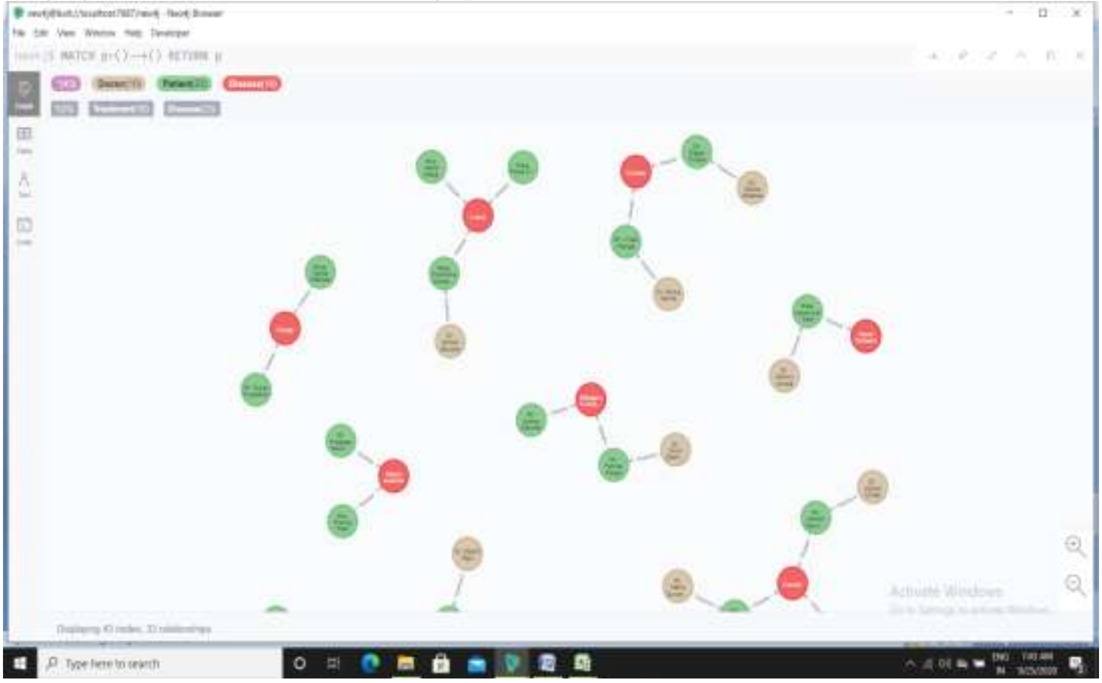
Figure 6.11 shows Single Node (Disease) Graph



```
MATCH (n:Disease) RETURN n LIMIT 25
```

Fig. 6.11 Single Node (Disease) Graph

Following figure 6.11shows the execution of graph of different node (Doctor, Patient &Disease)



```
MATCH p=()-->() RETURN p
```

Fig. 6.11 Execution of Graph of Different Node (Doctor, Patient & Disease)

Figure 6.12 showsNEO4J Browser after Graph Creation , it depicts after graph creation hospital database has 43 nodes and 33 relationship

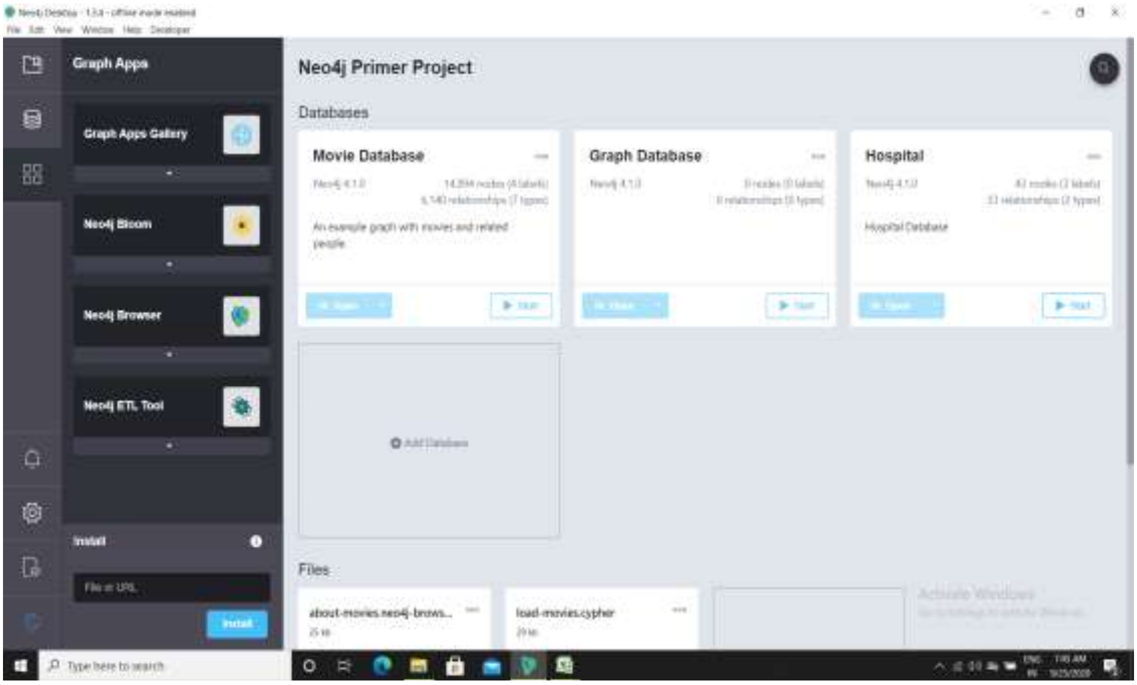


Fig. 6.12 NEO4J Browser after Graph Creation

6.2 Hospital Application Details

Hospital Information System are popular to deal with expanding populace needs and furthermore helps the rehearsing specialists and clinic administration and care staff with opportune assistance and accuracy. There are fluctuated measurements accessible to evaluate the exhibition of administrations like emergency clinic industry, and the fruitful execution and utilization of Hospital data framework shapes a critical role

.Hospital Applications are accessible in the product market which by and large should be redone and at times hospital information system should be created as an altered programming dependent on explicit emergency clinic necessities (client prerequisites).

As long as each stage implementation needs to be accurate and explicit, the hospital management system provides certain automation of many vital daily processes. The hospital application covers the administrations that bring together and rearrange crafted by medical services experts just as their associations with patients.

Figure 6.2.1 shows the Home page for hospital information system. From this page administater can log in after entering required data . This page has 5 fields namely Home , MCI, Hospital , Patient, Medical – care.

1]Homepage

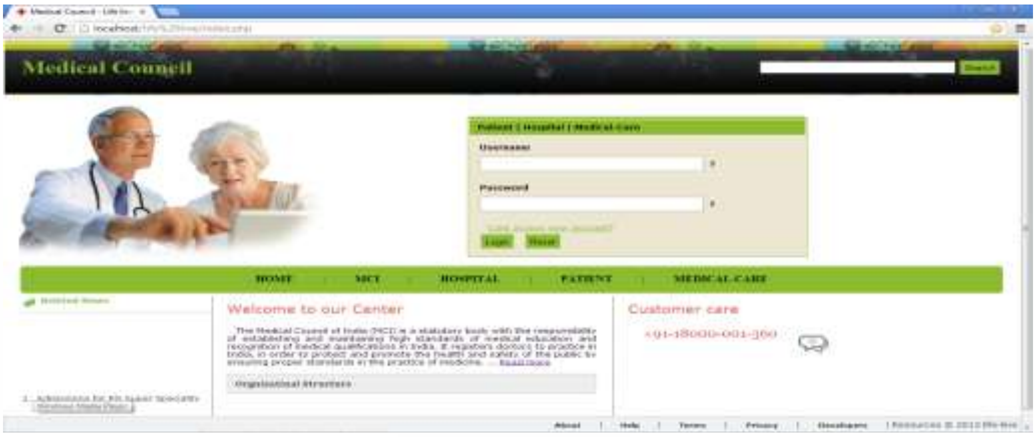


Fig 5.2.1 Home page

2) Doctor Login

Fig 6.2.2Shows the Doctor log in page. Here doctors needs to enter their username and password. If it is correct then they will get access to this software otherwise not.This Log in module is an integrated doctor management system, which captures complete and relevant doctors information. The system automates the doctors administration functions to have better and efficient service process. With this module doctor can register their name and login as well.

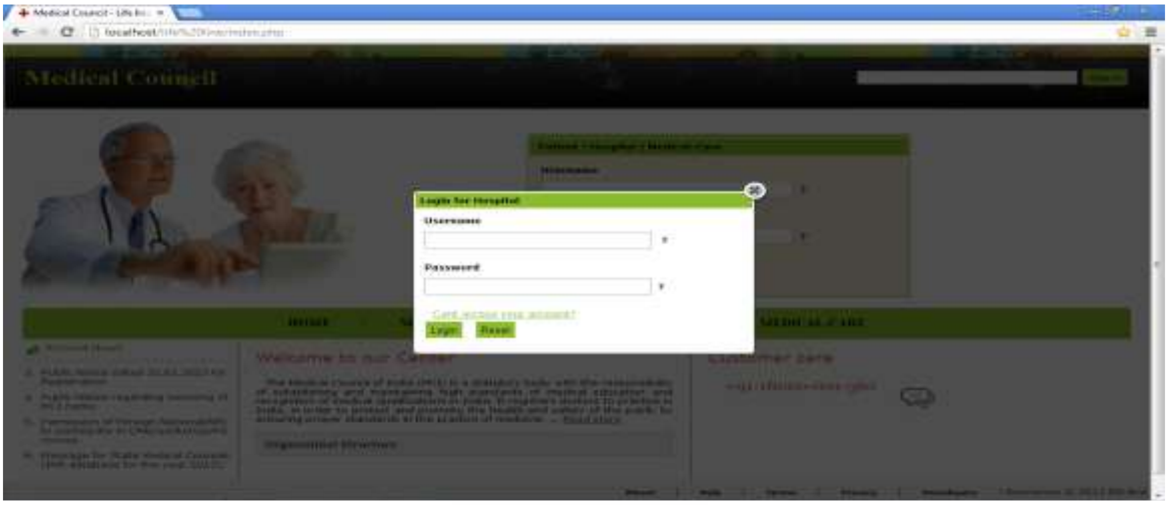
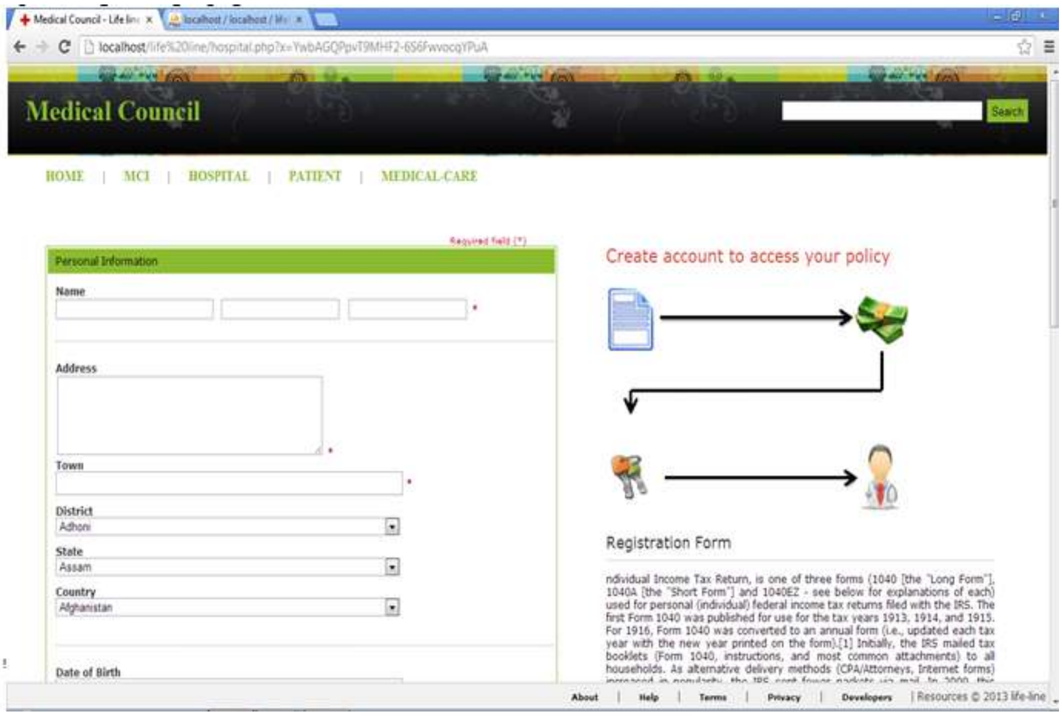


Fig 6.2.2 Doctor Login

3] Doctor Registration

Following figure shows the Doctor Registration page. This Registration module is an integrated doctor management system, which captures complete and relevant doctors information. The system automates the doctors administration functions to have better and efficient service process. With this module doctor can register their name and login as well. From this page doctor can register by entering required data.



localhost/life%20line/hospital.php?x=YwbAGQPpvT9MHF2-6S6FwvocqYPuA

Required field (*)

Personal Information

Name *

Address *

Town *

District: Adhoni

State: Assam

Country: Afghanistan

Date of Birth

Create account to access your policy

Registration Form

ndividual Income Tax Return, is one of three forms (1040 [the "Long Form"], 1040A [the "Short Form"] and 1040EZ - see below for explanations of each) used for personal (individual) federal income tax returns filed with the IRS. The first Form 1040 was published for use for the tax years 1913, 1914, and 1915. For 1916, Form 1040 was converted to an annual form (i.e., updated each tax year with the new year printed on the form).[1] Initially, the IRS mailed tax booklets (Form 1040, instructions, and most common attachments) to all households. As alternative delivery methods (CPA/Attorneys, Internet forms) increased in popularity, the IRS sent fewer packets via mail. In 2000, this

Medical Council - Life line
localhost / localhost / life
Medical Council
Search
HOME | MCI | HOSPITAL | PATIENT | MEDICAL-CARE
About | Help | Terms | Privacy | Developers | Resources © 2013 life-line

Fig 6.2.3 Doctor Registration

4] Waiting for login page

Figure6.2.4 shows waiting for login page

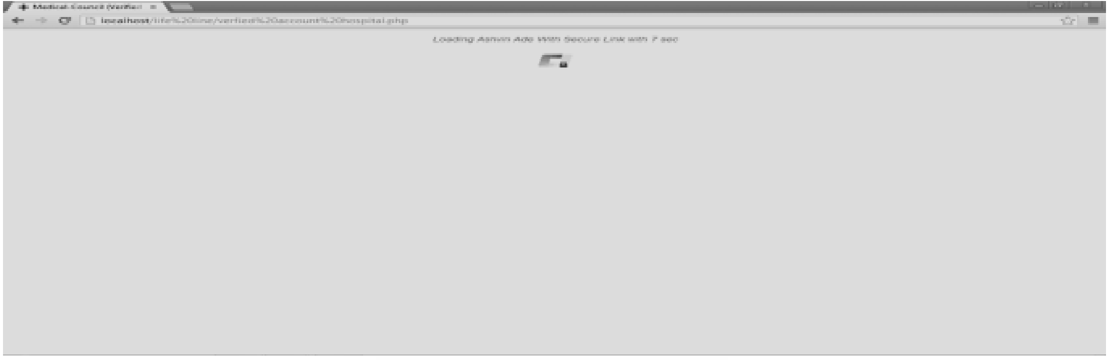
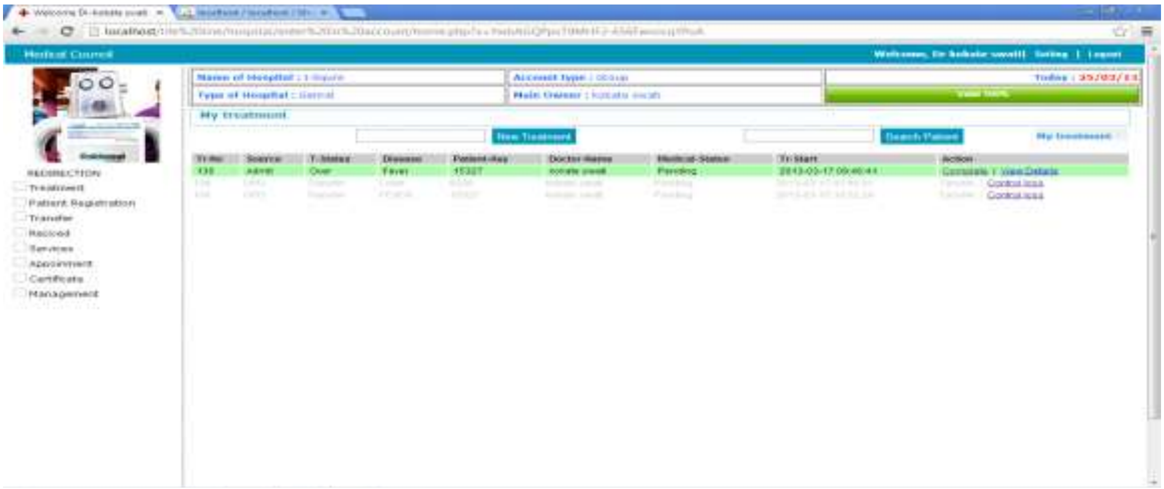


Fig 6.2.4 waiting for login page

5] Doctor login page

Figure 6.2.5shows the Doctor login page. It has information about the patient which doctors are treated with diseases. From this page doctor can get the information about his patient . It contains patient information with disease, patient admision date, patient contact number, address, paid fees etc.

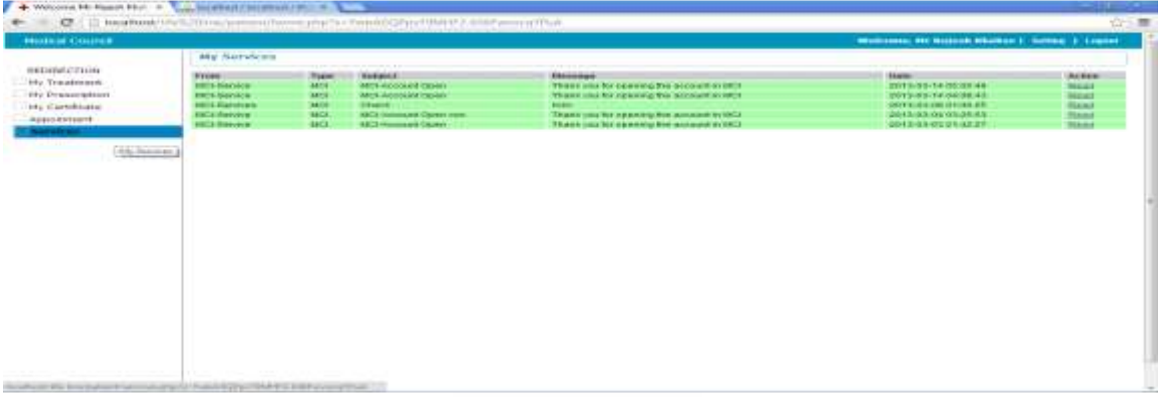


Tr.No	Source	T-Status	Disease	Patient-Key	Doctor-Name	Medical-Status	Tr.Start	Action
138	Admin	Over	Fever	15327	kotate swati	Pending	2013-03-17 09:40:41	Complete | View Details
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	Pending	[illegible]	[illegible] | Control loss
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	Pending	[illegible]	[illegible] | Control loss

Fig 6.2.5 Doctor login page

6] Patient login

Following figure shows the Patient login page. It contains information about patients.

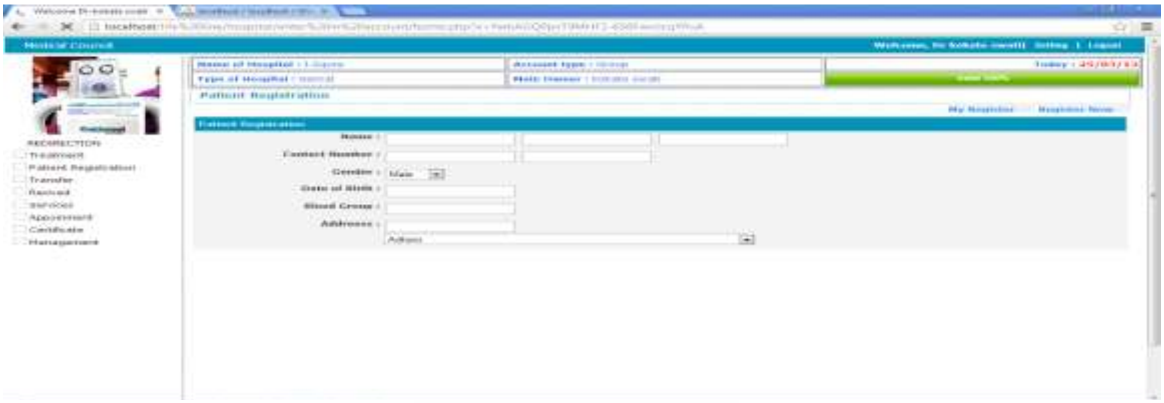


From	Type	Subject	Message	Date	Action
MCI-Service	MCI	MCI-Account Open	Thank you for opening the account in MCI	[illegible]	Read
MCI-Service	MCI	MCI-Account Open	Thank you for opening the account in MCI	[illegible]	Read
[illegible]	MCI	[illegible]	[illegible]	[illegible]	Read
MCI-Service	MCI	MCI-Account Open	Thank you for opening the account in MCI	[illegible]	Read
MCI-Service	MCI	MCI-Account Open	Thank you for opening the account in MCI	[illegible]	Read

Fig 6.2.6 Patient login

7] Patient Registration

Figure 6.2.7shows Patient Registration Form. From this page patient can register. This Registration module is an integrated patient information system, which captures complete and relevant patient information. The system automates the doctors administration functions to have better and efficient service process. With this module patient can register their name and login as well.



Medical Council

Patient Registration

My Register | Register Now

Patient Registration

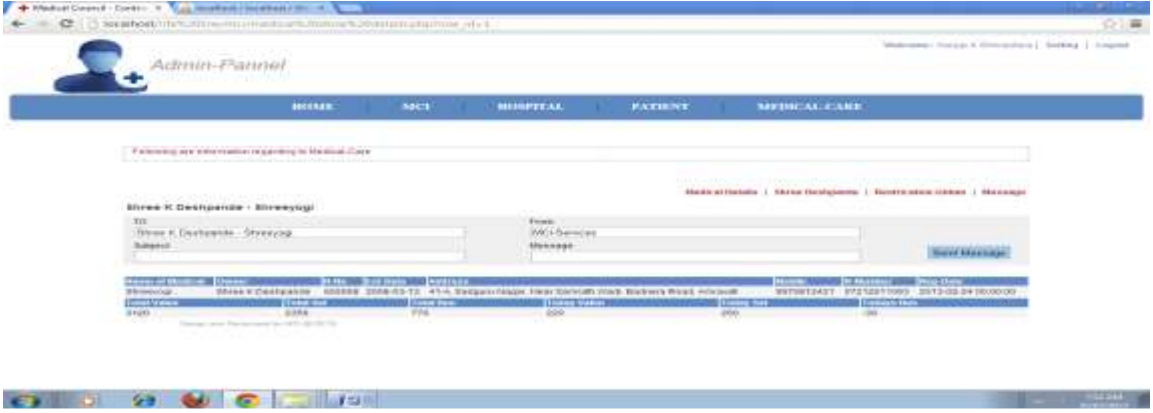
Name :
Contact Number :
Gender : Male
Date of Birth :
Blood Group :
Address :

REDIRECTION
Treatment
Patient Registration
Transfer
[illegible]
Services
Appointment
Certificate
Management

Fig 6.2.7 Patient registration

8] MCI Page

Following figure shows MCI page



Following are information regarding to Medical-Care

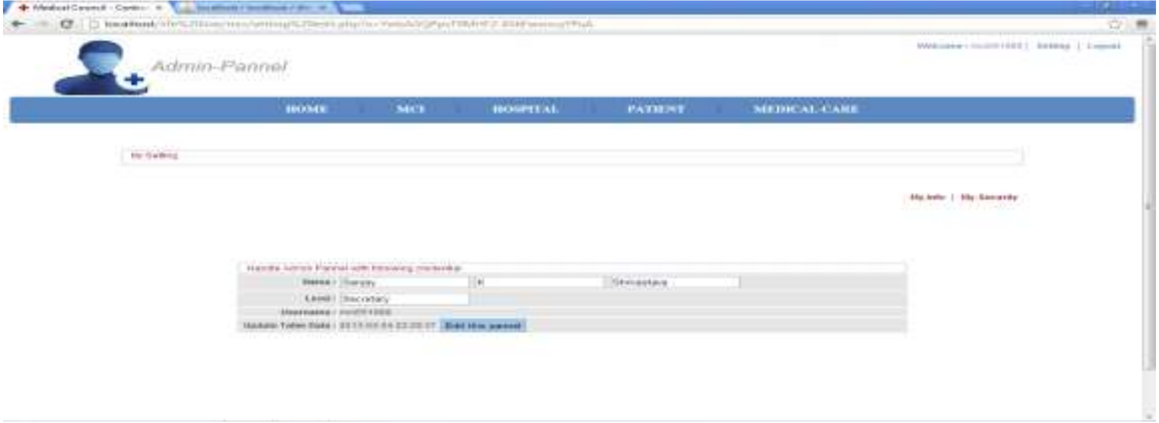
Shree K Deshpande - Shreeyogi

To: Shree K Deshpande - Shreeyogi
Subject:
From: MCI-Services
Message:
Send Message

Fig 6.2.8 MCI page

9] Update data

Figure 6.2.9 shows update data page. From this panel we can update the data.



My Setting

My Info | My Security

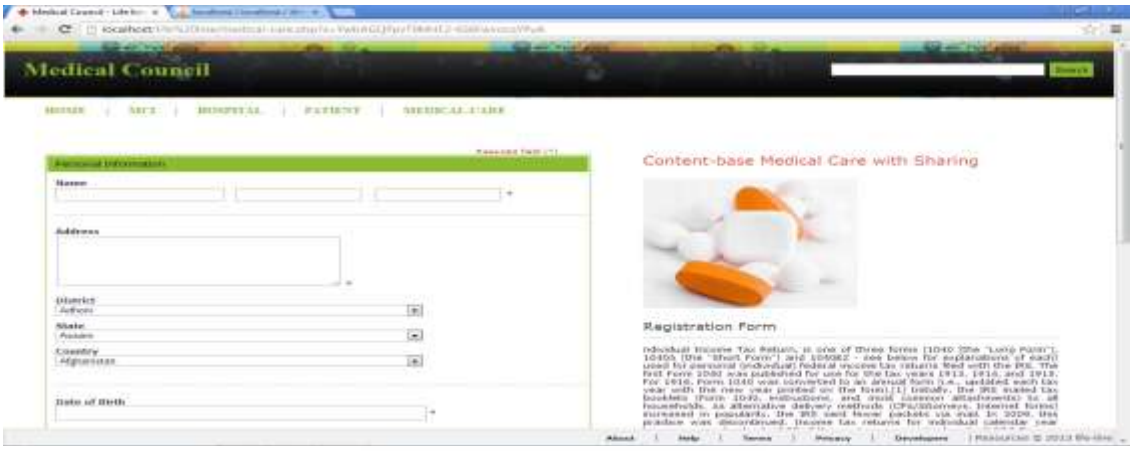
Handle Admin Pannel with following ownership

Name : Sanjay K Shrivastava
Level : Secretary
Username : [illegible]
Update Taken Date : [illegible]
Edit this panel

Fig 6.2.9Upadate data

10] Medical care registration

Following figure shows medical care registeration. From this page pharmacist can register. This Registration module is an integrated pharmacist, medicine, relevant patient information system, which captures complete and relevant patient, relevant medicine information. The system automates the doctors, patient administration functions to have better and efficient service process. With this module pharmacist can register their name and login as well.



ndividual Income Tax Return, is one of three forms (1040 (the "Long Form"), 1040A (the "Short Form") and 1040EZ - see below for explanations of each) used for personal (individual) federal income tax returns filed with the IRS. The first Form 1040 was published for use for the tax years 1913, 1914, and 1915. For 1916, Form 1040 was converted to an annual form (i.e., updated each tax year with the new year printed on the form).[1] Initially, the IRS mailed tax booklets (Form 1040, instructions, and most common attachments) to all households. As alternative delivery methods (CPAs/attorneys, Internet forms) increased in popularity, the IRS sent fewer packets via mail. In 2009, this practice was discontinued. Income tax returns for individual calendar year

Fig 6.2.10 Medical care registration

VII. CONCLUSION AND FUTURESCOPE

7.1 Conclusion

We compared the performance of a relational database (implemented inMySQL) and a graph database (implemented in Neo4j), in the context of hospital information system. It indicate that Neo4j performs better than relational database in queries that require multiple join operations when data size is 100.000 entries per table or node type.

Neo4j showed better results in terms of time ,response, compared with relational databases. Neo4j is best reasonable to execute complex organization of connections having a huge number of nodes and relationship.

7.2 Futurescope

Future work concerns deeper analysis of relational database and graph database. The work can be further extended to enhance the performance of the system. In general, the neo4j graph database can be better at the structural type queries than the relational database. In full-text character searches, the graph databses can be perform significantly better than the relational database.

There are four aspects that should be solved in order to achieve more conclusive results. The quality of the data impacts in the results because if Neo4j is a feasible alternative for replacing database models, the comparison executions should use real datasets. . As a second point, the operations of Update and Delete data should be evaluated and compared because there are very few databases that are only used for insertions and readings. Therefore, scientific evidence of those operations are mandatory. After that, in enterprise contexts operative systems like UNIX, Solaris and Linux Red Hat are more common. It is recommended to compare the performance in more commercial operating systems.

A future comparison between Neo4j and relational model should evaluate the most popular relational databases like Oracle, SAP, Microsoft and IBM.

REFERENCES:

[1]. S. Sencer and K. Eren. "Graph Database for Agent Based Emergency Response Model," Proceedings of the 2014 international conference on advances in big data analytics: July 21-24, 2014.

[2]. K. Nagi. "A New Representation of WordNet using Graph Databases" The fifth international conference on advances in databases, knowledge, and data applications. 2013:1-8

[3]. J. Webber. "A Programmatic Introduction to Neo4j," 2012 ACM conference on systems, programming, and applications: software for humanity, October 19-26, 2012, Tucson, Arizona.

[4]. Y. L. Wang. “A comparative study of graphic database NEO4J and relational database,” Modern Electronic Technology, vol. 35, pp. 77-7, 2012.

[5]. L. Wang. “Construction of Document Resources Association Network Based on graph database technology,” Digital Library Forum, pp. 59-65, 2014.

[6]. Y. L. Wang. “Research on embedded application of graphic database,” Neo4j modern electronic technology, vol. 35, pp. 22, 2012

[7]. B.T. Messmer et al, D. Montag, “Understanding Neo4j Scalability,”NeoTechnology, USA, 2013.

[8]. H. Darwen, An Introduction to RelationalDatabase Theory, United Kingdom: Bookboon. com, 2010.

[9]. J. Zhang, Z. Li and S. Liu, “ Graph Search with Neo4j,” Department of Computer Science, Georgia State University, Atlanta,USA, 2013.

[10]. Pere Burton , R. Angles, A. Prat-Pérez, D. Dominguez-Sal and J.-L. Larriba-Pey, “Benchmarking database systems for social network applications,” in First International Workshop on Graph Data Management Experience and Systems, NewYork, USA, 2013

[11]. C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y.Chen and D. Wilkins, “A Comparison of aGraph Database and a Relational Database,” inACMSE 10’, Oxford, MS, USA, 2010.

[12]. H. Florian and P. René, “Performance of Graph Query Languages: Comparison of Cypher,Gremlin and Native Access in Neo4j,” ACM, pp.18-22, 2013.

[13]. WisalKhan,Ejazahmed,WaseemShahzad, “Predictive Performance Comparison Analysis of Relational &NoSQL Graph Databases,”National University of Computerand Emerging Sciences,Islamabad, Pakistan, IJACSA,Vol**8,** No.**5**,2017

[14]. Mahmut, Yusuf Kavurucua, AlevMutlubMahmut Idea, Yusuf Kavurucua, AlevMutlub, “Graph Representation of Relational Database for Concept Discovery”, aTurkish Naval Academy ,Computer Engineering, Istanbul-34940, TürkiyebKocaeliUniversity,Computer Engineering,Kocaeli-41000, Türkiye, Science Direct Procedia - Social and Behavioral Sciences **195 (2015) 1981 – 1989**

[15]. Jaroslav pokorny,michalvalanta,jirikovacic, “ Integrity constraints in graph databases”, Faculty of Mathematics and Physics, Charles University, Malostranskénám. 25, 118 00 Prague, Czech Republic bFaculty of Information Technology, Czech Technical University, Thákurova 9, 160 00 Prague, Czech Republic, ScienceDirectProcedia Computer Science **109C(2017)975-981**

[16]. David W. Williams, Jun Huan, Wei Wang, “Graph Database Indexing Using Structured Graph Decomposition”, 1Department of Computer Science University of North Carolina, Chapel Hill 2Department of Electrical Engineering and Computer Science University of Kansas.

[17]. ShaliniBatra, CharuTyagi, “Comparative Analysis of Relational And Graph Databases”,IJCSE ISSN: **2231-2307**,Volume-**2**,Issue-**2,**may**2012**.

[18]. Renzo Angles And Claudio Gutierrez.” Survey Of Graph Database Models”ACM Computing Surveys, Vol. 40, No. 1, Article 1, Publication date: February 2008.

[19]. Renzo Angles. “Comparison of Current Graph Database Models”, Department of Computer Science, Talca.

[20]. HuilingLu ,Zhiguo Hong , Minyong Shi, Communication University Of China, “Analysis Of Film Data Based On Neo4j”

[21]. Karamjit Kaur and RinkleRanin, ThaparUniversity , “Managing data in healthcare information system: Many Models One Solution”