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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

Effective Handling of Big Data with NoSQL

S.Suguna Devi, Vijayan, R. Ravi Kumar,

M.E, Dept. of CSE, Rrase College of Engineering, Chennai, TN, India

Assistant Professor, Dept. of CSE, Rrase College of Engineering, Chennai, TN, India

HOD, Dept. of CSE, Rrase College of Engineering, Chennai, TN, India

ABSTRACT: Big Data is a phrase which refers to a large volume and Wide variety of data being captured from different sources at high speed. It is estimated that data volume is increasing 40% per year, and will grow 44 times between 2009 and 2020. Much of this data is of a textual nature and hence unstructured. With the emergence of Big Data, the use of No SQL technology is rising rapidly among internet companies and the enterprise. Benefits include simplicity of design, horizontal scaling and finer control over availability. No SQL databases are increasingly considered a viable alternative to relational databases, as more organizations recognize that its schema less data model is a better method for handling the large volumes of structured, semi structured and unstructured data, being captured and processed today. For example No SQL databases are often used to collect and store social media data. The purpose is to introduce the concepts, highlight the different No SQL database types, and provide arguments for and against adopting No SQL. No SQL databases are non-relational and accommodate unstructured data. It is not a replacement for an SQL database but compliments it; both technologies can coexist. The key difference is that relational (SQL) databases have rigid schemas while No SQL databases offer a flexible schema design that can be altered without downtime or service disruption. No SQL was also designed for distributed data stores for large scale data needs; for example Face book has 500 million users and Twitter accumulates terabytes of data. No SQL takes advantage of scaling out, by spreading its load over many database servers, and this is an inexpensive solution for large datasets.

KEYWORDS: Big Data, No SQL, Relational Database, Schemas, Distributed System.

I. INTRODUCTION

No SQL databases are increasingly considered a viable alternative to relational databases, as more organizations recognize that its schema less data model is a better method for handling the large volumes of structured, semi structured and unstructured data, being captured and processed today. For example No SQL databases are often used to collect and store social media data. The purpose is to introduce the concepts, highlight the different No SQL database types, and provide arguments for and against adopting No SQL. No SQL databases are non-relational and accommodate unstructured data. It is not a replacement for an SQL database but compliments it; both technologies can coexist. The key difference is that relational (SQL) databases have rigid schemas while No SQL databases offer a flexible schema design that can be altered without downtime or service disruption. No SQL was also designed for distributed data stores for large scale data needs; for example Face book has 500 million users and Twitter accumulates terabytes of data. No SQL takes advantage of scaling out, by spreading its load over many database servers, and this is an inexpensive solution for large datasets. However in comparison to relational databases, No SQL databases do not have the same distinct properties or data integrity. In spite of this No SQL databases have an established track record for handling Big Data efficiently. The primary way in which No SQL databases differ from relational databases is the data model; there are numerous No SQL databases and they primarily fall into the following four categories: Key Value Databases use a hash table where there is a unique key and a pointer to a specific set of values; data can only be queried by the key. The data can be unstructured, as it does not enforce a set schema across key value pairs. Face book uses this database type, as the datasets are not related to each other and the data is unstructured. The simplicity of this database type makes it ideally suitable for fast highly scalable retrieval of values needed for application tasks like managing customer profiles and retrieving product names. Amazon uses its own key value database, Dynamo DB, for its shopping cart. These databases were built for quick and efficient data management in distributed systems.

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

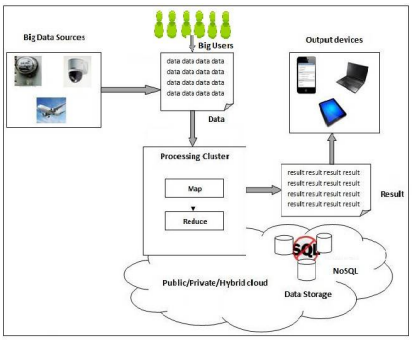
International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

II.RELATED WORKS

Out of the many different data-model architectures, the relational data model architecture has been dominating since the 80s, with the implementations like Oracle database [3], MySQL [4] and Microsoft SQL Servers. Laterly, however, the relational databases leads to the problems in many cases because of its data modeling techniques.The exponential growth of complexity of data generated by social networks, sensors, real time systems, and global users etc, and the storage of this huge amount of data on big distributed system, demands evolution of new data management model .



III. ORGANIZATIONS MIGRATING TOWARDS NOSQL DATABASE

NoSQL databases focus on analytical processing of large scale datasets in warehouses, offering increased scalability over commodity hardware and servers. Computational and storage requirements of applications such as for Big Data Analytics, Business Intelligence and social networking over peta-byte datasets have published SQL-like rapidly, and the nature of data is changing as well.The various kinds of data is collected and it demands for a very different type of database which should be very flexible and easily incorporate any new type of data. So the database must have a capability of efficiently storing and very fast access to the new types of data that includes semi-structured and unstructured data.Unfortunately, the relational databases have very poor features to quickly adopt new types of data because of its rigid and static schema based approach, and is not suitable for semi-structured and unstructured data. Finally, the NoSQL meets the growing trends of storage, processing and retrieval of data by providing a flexible, schema-less data model that maps the organization's requirement and simplifies the communication between the application and database, that results in less writing code, debugging and maintenance becomes more easier. The strict relational schema of relational databases can be a burden for web applications like blogs, which consists of much different kind of attributes. Text, audios, pictures, videos, real time data and other fast changing information have to be stored within multiple tables. Since such web applications are very agile, underlying database have to be flexible and dynamic as well in order to support easy schema evaluation process [18]. NoSQL systems exhibit the ability to store and index arbitrarily Big Data sets while enabling a large amount of concurrent user requests.

A. KEY VALUE STOREDATABASES

These are the simplest NoSQL databases. It helps developers to build applications with schema-less, unformatted data.

B. COLUMN ORIENTED DATABASES

These database stores their data in the form of columns, making it faster read a particular column to memory and making calculations on all values in a column.These are optimized for queries over large datasets, and stores column of data together.

Example: Cassandara [29], Hypertable [30] etc…

C. DOCUMENT ORIENTED DATABASES

These databases make use of JSON or XML format to store the values which is then called as document. These databases support complex data structures and helps in easy debugging, conceptualizing data.

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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

Ii.UNLIKE RELATIONAL DATABASES WHICH STORES AND RETRIEVE DATA FROM INTERRELATED TABLES, DOCUMENT DATABASE CAN STORE AN ENTIRE OBJECT IN A SINGLE JSON DOCUMENT, MAKING IT FASTER TO RETRIEVE

In comparison with relational databases, document databases support in freely addition of fields to JSON documents, no need to define changes initially. And also these databases support dynamic data that can be changed at any time.

D. SECURITY CHALLENGES IN NOSQL DATABASES

The NoSQL databases emerge with different security issues. The main focus of NoSQL databases is handling the new data sets, with less priority on security. The NoSQL databases are built to meet the requirements of analytical world of big data, and less emphasis on security is given during design stage. NoSQL databases do not provide any feature of embedding security in the database itself. Developers need to impose security in the middleware.

IV. PROPOSED SYSTEM

In the Proposed system, we are building Big Data Application with No SQL Data. We are implementing Mango DB for our Application Deployment. Each record and its associated data are usually stored together in a single document; this simplifies data access and reduces the need for joins or complex transactions. Documents stored are schema free and similar to each other; this flexibility can be particularly helpful for modelling unstructured data.

V. TRANSACTIONAL INTEGRITY

NoSQL databases are failed to ensure transactional integrity because of its soft nature. Complex integrity constraints cannot be added in NoSQL database architecture because it results in failure to meet the NoSQL's main objective of attaining better performance and scalability.

VI. AUTHENTICATION MECHANISMS

NoSQL databases are exposing to replay attacks, password brute force attacks, cross-site request forgery, injection attack and man-in-the middle attack results in information leakage. The main reason is NoSQL databases incorporate the weak authentication mechanism and weak password storage techniques. Some NoSQL databases enforce authentication mechanism at local node level, but fail to enforce authentication across all commodity servers.

VII. SUSCEPTIBILITY TO INJECTION ATTACKS

Injection attacks add its own choice of data to the NoSQL database results in unavailability and corrupted data. Since NoSQL employs very light weight protocols and loosely coupled mechanism in its architecture that allows an attacker to backdoor access of a file system for malicious activities.

a) LACK OF CONSISTENCY

b) NoSQL databases does not satisfies simultaneously all the three properties (consistency, availability, and partition fault tolerance) stated by CAP theorem. NoSQL databases make use of many distributed commodity servers, it does not assure consistent results at all time, as all participating commodity servers may not entirely synchronized with other servers holding latest information. If a single commodity server gets fail, results in load imbalance among other commodity servers.

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

International Journal of Innovative Research in Computer and Communication Engineering

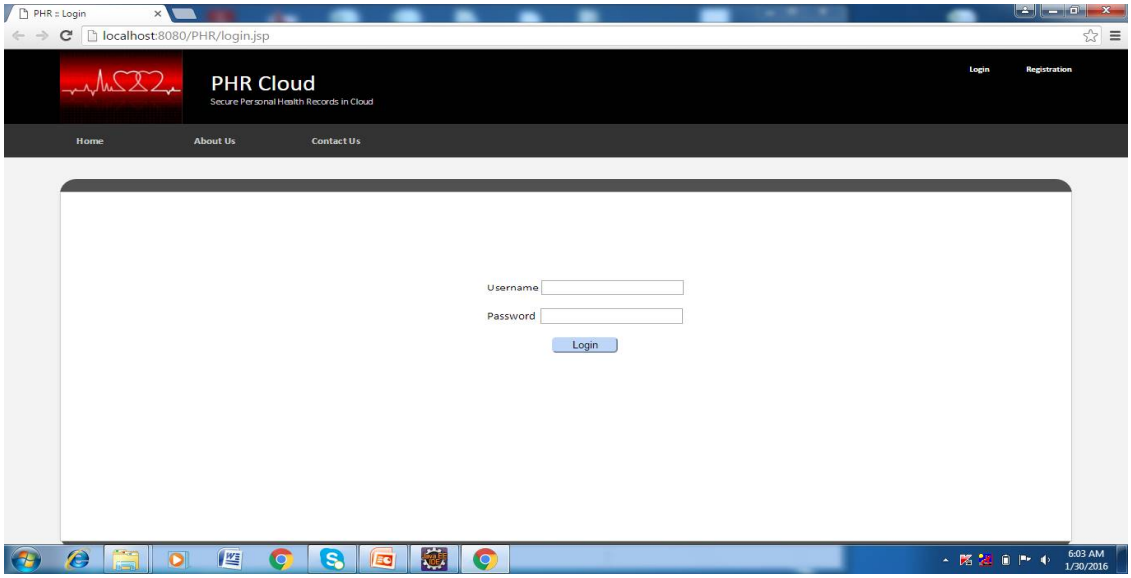
(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

c) **INSIDER ATTACKS:**

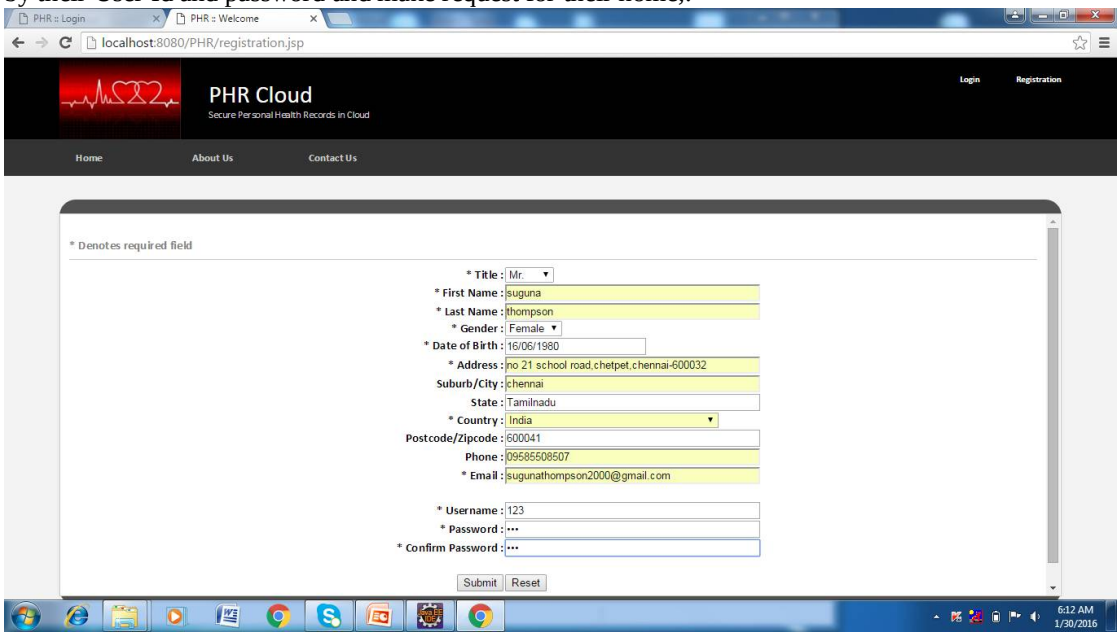
d) NoSQL databases has poor logging and log analysis methods, due to this an insider attack can gain access to critical data of other users. As NoSQL databases has very thin security layer, it becomes very much difficult for users to maintain control over their data. NoSQL databases make use of many distributed commodity servers, it does not assure consistent results at all time, as all participating commodity servers may not entirely synchronized with other servers holding latest information. If a single commodity server gets fail, results in load imbalance among other commodity servers.

VIII. RESULTS



A.HOME PAGE

This is the homepage consist of the user name and password. In the Login module mobile user can login by their User Id and password and make request for their home,.



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

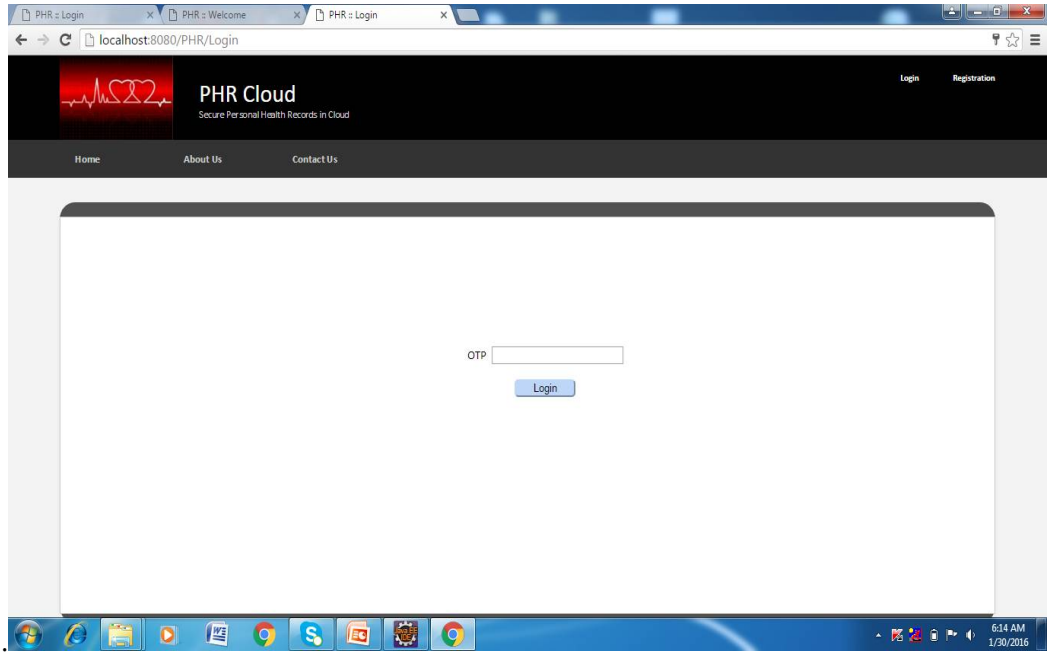
International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

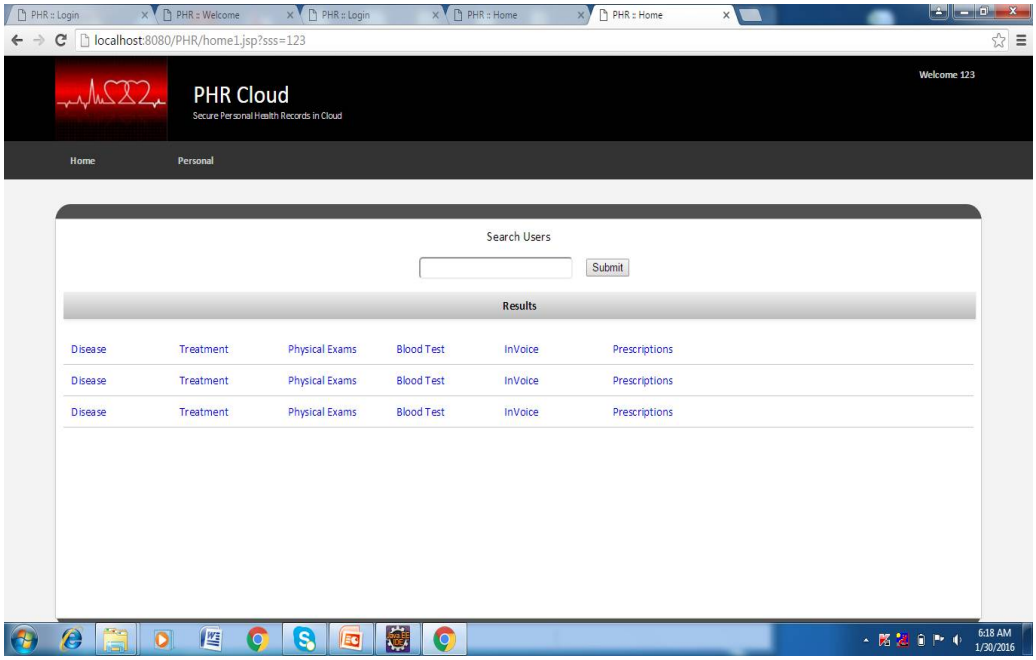
B.REGISTRATION PROCESS

In this Module every user will be Register with the Server. So user has to give User Name, Password, Address, Mobile number and other details



C.TOKENENIZATION

In this we create a code to retrieve the information from the database here we create our database in db. Tokenization technique is used.



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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 5, May 2016

D.DETAILS OF THE PATIENT

In this module we partition the data and store in the different database, we use db to store all the data, db is database as service system in the cloud so we need to install any softer for storing the data, every data in the db are in the descriptive.

IX. CONCLUSIONS

- NoSQL databases provides schema-less dyanamic flexible data model, that is most suitable for the big users and big data.
- NoSQL databases have an ability to scale dramatically to support global users and big data.
- NoSQL databases provide an improved performance to satisfy big users expectation without compromising scalability.

X. FUTURE ENHANCEMENT

Both prototype systems effectively address the libraries data storage requirements, but differ in their approach. APEX requires table structure to be defined before adding data whereas Mongo DB does not explicitly create collections; document structure is defined automatically during the first data insert. Mongo DB enables the library to either embed product attributes into a single document or reference the data in another document. However with a normalized SQL database, product attributes might be spread over numerous tables; therefore complex SQL joins are required, to view all the attributes of a product. Also referential integrity rules specify that the libraries SQL database should only insert records with Author ID into the Product Book table, if the Author ID exists in the Author table. The No SQL database is very efficient and allows the library to obtain all film attributes using one simple query.

REFERENCES

[1]. Anna Bjorklund, ―NoSQL databases for Software Project data‖. January 18, 2011.
[2]. Find Couchbase from http://www.couchbase.com
[3].Oracle Databases from web: http://www. oracle.com/us/products/database/overview/index tml
[4]. MySQL Databases from web: http://www.mysql.com/
[5]. Microsoft SQL Server Databases from web: http://www.microsoft.com/en-us/sqlserver/default.aspx
[6]. A B M Moniruzzaman and syed Akhtar Hassan, ―NoSQL database: New Era of databases for Big data Analytics-Classification, Characteristics and comparison.‖
[7]. Levih, N (2010). ―Will NoSQL databases live up to their promise? ‖