ANALYSIS OF STAND-ALONE AND CLOUD-BASED MONGODB FOR DATASET MANAGEMENT

NAME: SANTTOSH A/L MUNIYANDY
MATRIC NUMBER: 159193
COURSE CODE: CPM451
STUDENT EMAIL: santtosh3476@student.usm.my

NAME:ZHANG YUBIAO
MATRIC NUMBER:155062
COURSE CODE:CPC451
STUDENT EMAIL: yukayukazhang233@student.usm.my

NAME: MUHAMMAD ASHRAFF BIN SHAFIE
MATRIC NUMBER:155405
COURSE CODE: CPM451
STUDENT EMAIL: ashraffshafie@student.usm.my

NAME: ZHANGJIAMAN
MATRIC NUMBER:155118
COURSE CODE:CPC451
STUDENT EMAIL: zhangtop@student.usm.my

***Abstract*— Several indications support that Big data technology will be widely used in the healthcare industry in the near future. Therefore, the study develop several analysis methods to test the performances of stand-alone MongoDB and MongoDB atlas which have high potential in healthcare industry. Stand-aloneMongoDB database shows higher potential to be adapted in healthcare industry as compared to MongoDB atlas.**

I. Introduction

Big Data technologies are defined based on the characteristics of 3V which are Variety, Velocity and Volume or more V's [4]. These characteristics are crucial to cater to the amount of data created in this digitalisation era.

Healthcare is a complex, multi-dimensional system aimed at preventing, diagnosing, and treating health-related issues in humans. The procedures involving these professional includes primary care such as consultation to quaternary care for rare diagnostic or surgical procedures. Each level requires different types of information management, including patient medical history, clinical data from imaging and laboratory tests, and other personal medical data. The main challenge in healthcare information management system is the time required to make a decision for an increasing number of patients as human population is increasing.

Gene therapy is one of the latest technology in medicine, designed to treat genetic disease. The technology involved taking sample from the newly emerged Deoxyribonucleic Acid (DNA) of the baby in the womb when it was still in the zygote stage. The procedure was then proceed with DNA sequencing and the challenge come because the DNA of human genome is too large, which is 3Gbp or around 0.75 GB per patient [6]. The turnover time for the result to be analysed is around 1-4 days depending on the processing technology which is used just for one patient [5]. Therefore, the use of big data technology might help to reduce the time taken for the result to be produced in a much shorter time. This in turn will make gene therapy to be executed earlier to the baby before the cells had divided into several cells which make it harder to repair many cells as compared to fewer cells.

MongoDB is one of the current big data technologies that is a document based model, which offer a cloud-based platform (MongoDB Atlas) and also a local-based platform (stand alone MongoDB). Our objective is to analyse the performance of these two platform of MongoDB to be used in healthcare industry for screening-diagnostic purposes.

II. ANALYSIS OF SELECTED DATASET

. The dataset that was chosen for this study was the Behavioral Risk Factor Surveillance System (BRFSS) which was obtained from [1]. In 1984, the Centers for Disease Control and Prevention (CDC) established these surveys to collect data yearly from the residents of the United States. This survey concerns various health risks and conditions. [2]. The data of this survey is conducted by telephone interviews with more than half a million participants each year covering all over the states of the United States. The variable definitions of these are available from : https://www.cdc.gov/brfss/annual_data/2015/pdf/codebook15_llcp.pdf. The large size of these data provides meaningful insights depending on the total number of records and all the

variables. The reason for this dataset being chosen is that it offers numerous health-related data which covers a wide range of health behaviors thus, making it applicable for many research and educational purposes. Moreover, this dataset is the size of 512MB for each year of data which allows many complex queries capabilities that can be carried out in MongoDB.

The variable that was utilized in this project:

a) SMOKDAY2: Smoking cigarettes everyday
b) SEX: Gender
c) _PACAT1:Physical Activity categories (Levels of Active)
d) _AGE_G: Age Category (e.g: 18-24, 25-34)
e) GENHLTH: State of our health (e.g: Excellent, Poor)
f) MARITAL: Marital Status
g) ADDEPEV2: Have depressive disorder
h) _DRNKWEK: Total times alcohol consumption per week
i) CHILDREN: Total number children in household
j) _STATE: States in U.S

III. PROJECT METHODOLOGY

In this study, we have selected two platforms for MongoDB to implement our queries and evaluate their performances which are stand-alone MongoDB and cloud-based MongoDB (MongoDB Atlas). The documents we referred to for our implementation is The little MongoDB Book as it contains every set of manual and aid needed from creating clusters and databases to guides of query[8].

Without Further Ado, the process of installation for stand-alone MongoDB was involved downloading from its official website: https://www.mongodb.com/try/download/community-kubernetes-operator . After installing the application according to our operating system which in our case is Windows, we will be given the option to connect to a new connection for our local host device. Once connected to our new connection, we simply head to databases and press 'create database' to begin importing our BRFSS into it. Once inserting the new name for the database and the collection, all that we need to do is select our database from our documents and import it in. Although, the process of importing depends on the size of our database which in our case it wasn't relatively too quick. Once it has been uploaded into our collection, we are free to query and aggregate around the database.

In the case of MongoDB Atlas, this does not require installation as we head to its official MongoDB atlas website: https://www.mongodb.com/products/platform/atlas-database and log in with our account. After logging into our Atlas account, we need to create a cluster where there will be 2 paid options and a free tier option. For the case of a not too large dataset, we utilize the free tier option. To implement our chosen dataset, we are needed to connect our cluster on Atlas with our Compass, which requires installation for those who have not installed within their device. Although, this is just one of the methods that they have provided. We will be given a connection string where we need to fill in certain parts such as our username and our password which was created during the process of creating a new cluster. Paste this connection string within our url section, which then opens our installed compass and connects automatically. Thus, we are finally able to import our dataset by the same procedure as mentioned before in the MongoDB stand-alone section. Once imported our dataset, head back to Atlas and we are able to query and aggregate to our wishes. The only downside to Atlas is that if we are utilizing free tier clusters, we are limited to 512MB as maximum dataset, which opposes compass such that we can upload any size depending on our local device storage space.

After uploading the dataset on both platforms, we are finally moving onto implementing our queries to evaluate their performances. The four meaning queries that we will be running is:

1. To retrieve the total documents of women who smoke every day:

   { SMOKDAY2: 3, SEX: 2 }

2. To retrieve the total documents of men between 18 and 24 who are highly active in their daily life but still suffer poor health:

   {_PACAT1:1,SEX:1,_AGE_G:1,GENHLTH:5}

3. To retrieve the total documents of both men and women who drink below 100 times weekly and suffers depression in their daily life:

   {MARITAL:1,ADDEPEV2:1,_DRNKWEK:{$lt:100}}

4. Finally, for this part we decided to experiment and utilize the aggregation of MongoDB. Thus, this final query is: To calculate the sum of the documents of females between 25 to 34 who have more than 4 children and have depression in the state of California:

a) First section of this part we create the first stage of aggregation: MATCH:

{ CHILDREN:{$gte:4},SEX:2,ADDEPEV2:1,

_AGE_G:2,_STATE:6 }

b) Second section of this aggregation: GROUP:

{ _id: "$_STATE",

total: { $sum: 1}}

Comparison query operation to notice of:

Operation	**Description**
$eq	matches the values for equal to specified value
$gt	matches values for greater than specified value
$gte	matches values for greater than or equal specified value
$lt	matches values for lesser than specified value
$lte	matches values for lesser than or equal to specified value
$ne	matches values for not equal to specified value
$in	matches values for values specified in an array
$nin	matches for not equal to values specified in an array

[7]

IV. COMPARISON, DISCUSSION AND RECOMMENDATION

In using MongoDB Atlas and MongoDB Local, several differences emerge that are for users to consider based on their specific requirements. MongoDB Atlas, being a cloud-based platform, has no need for manual installation and offers scalability and ease of setup, making it great for startups and teams requiring remote collaboration. However, it imposes a data limit of 512MB in its free tier, which may not suffice for larger datasets, unlike MongoDB local, which is limited only by the local machine's capacity. From a performance perspective, local installations like MongoDB local might offer faster query execution due to the absence of network latency, which is a factor in cloud-based Atlas. Cost is another consideration; while Atlas may seem cost-effective at the outset, especially at the free tier, scaling resources can increase expenses. Conversely, MongoDB local involves initial setup costs and ongoing maintenance of physical hardware. Security-wise, Atlas provides managed security, which can be more appealing than managing security in-house with MongoDB local.

Aspect	**MongoDB Atlas (Cloud)**	**MongoDB (Local)**
Installation & Setup	No installation required; setup done via web interface.	Requires software download and installation on local machines.
Data Storage Capacity	Limited to 512MB in free tier; scalable with paid plans.	Limited only by local hardware capacity.
Performance	Potentially affected by network latency.	Faster access due to local data storage.
Scalability	Easily scalable without hardware restrictions.	Manual scaling; limited by physical hardware.
Cost	Free tier available; costs increase with scaling.	No ongoing costs except for local hardware and maintenance.
Security	Robust security managed by the provider.	Requires in-house security management.
Use Case Appropriateness	Best for startups, remote teams, and collaborative projects.	Suitable for enterprises with large, sensitive data needs.
Reliability & Accessibility	High reliability and accessibility from anywhere.	Dependent on local system reliability.
Best Practices	Optimize resource use to control costs, ensure regular monitoring.	Conduct regular backups and security audits.

V. CONCLUDING REMARKS

As can be concluded from the above, MongoDB Atlas is a subscription-based hosting service for MongoDB on servers, which has the advantages of faster configuration and expansion process and no need to maintain information infrastructure. It is recommended for users needing quick setup, scalability, and collaborative access, and is particularly suitable for small to medium-sized projects. For large

enterprises with extensive data needs and higher concerns for data control and security, MongoDB Local is advisable. In this case, using open source MongoDB does not require paying high licensing fees, but only requires focusing on maintaining system operations, allowing easy access to source code to improve MongoDB and even promote the development of the entire MongoDB community.[3]

VI. CONCLUSION

This project analyzes the performance and applicability of standalone and cloud-based MongoDB platforms for managing healthcare datasets. Our research focused on the Behavioral Risk Factor Monitoring System (BRFSS) dataset, which was selected for its comprehensive health-related data from U.S. residents.Standalone MongoDB provides complete data control and potentially faster data processing. It requires a lot of technical expertise and resources to set up, configure, and maintain. MongoDB Atlas provides a more user-friendly experience with quick setup, automated management capabilities, and scalability through a web interface, making it better suited for remote collaboration and projects that require rapid deployment。 MongoDB Atlas is faster to set up and easier to manage, but you may experience network latency that affects query speed. The free tier in MongoDB Atlas limits the data store to 512MB, while the standalone MongoDB is limited only by local hardware capacity, making it more suitable for larger datasets.MongoDB Atlas provides managed security, reducing the burden on internal teams, while standalone MongoDB requires internal management of security measures. In terms of cost, MongoDB Atlas offers a free tier, but it can get expensive as you scale, whereas standalone MongoDB has upfront hardware costs but no ongoing fees.

In conclusion, MongoDB Atlas is recommended for projects that require quick setup, scalability, and ease of management, and is suitable for startups and collaborative teams. Standalone MongoDB is better suited for enterprises with big data needs and the ability to manage their infrastructure. The choice between these platforms should be based on project requirements, available resources, and long-term goals, ensuring efficient and effective data management in the healthcare industry

VII. LESSON LEARNED

Throughout the project, we gained knowledge and lessons about managing healthcare datasets using the standalone and cloud-based MongoDB platform, and we learned about the diversity, speed, and volume of big data. Effective data management in healthcare requires a platform that can handle diverse, rapidly changing, and large-scale data sets. And for standalone MongoDB, there is no network latency, providing complete control over the data and potentially faster processing speeds. It requires a lot of technical expertise and resources to set up and maintain. MongoDB Atlas is easy to set up, scalable, and a managed service, making it ideal for projects that require rapid deployment and remote collaboration. But it can incur higher costs due to scaling and face network latency issues. The installation process for standalone MongoDB is more complex and time-consuming, involving multiple manual steps. MongoDB Atlas has a simplified web interface setup that saves time and reduces the need for a lot of technical skills. MongoDB Atlas' automated management features reduce operational burden. In contrast, standalone MongoDB needs to handle these tasks manually. While MongoDB Atlas provides the convenience of cloud-based access, query processing speed can be affected by network latency.

VIII. GROUP MEMBER'S ROLES

In this collaborative effort, all authors exhibited commendable responsiveness and cooperation. Santtosh A/L Muniyandy undertook the responsibility of drafting the highlights among the analysis of selected dataset, project methodology, and author contributions. Additionally, Muhammad Ashraff contributed significantly to elucidating abstract, while also playing a key role in introduction. Zhang YuBiao diligently addressed the lesson learned and conclusion. Zhang JiaMan focused on detailing the comparison, discussion and recommendation overall of our study based on utilizing both MongoDB platforms and concluding remarks.

IX. REFERENCES

1.Centers. (2015). *Behavioral Risk Factor Surveillance System*. Kaggle.com. https://www.kaggle.com/datasets/cdc/behavioral-risk-factor-surveillance-system/data

2. CDC. (2020). *CDC - About BRFSS*. Centers for Disease Control and Prevention. https://www.cdc.gov/brfss/about/index.htm

3.Baby, R. (2020, October 12). MongoDB Atlas database vs MongoDB local, which is best for SaaS in terms of transaction speed (Querying) [Forum post]. Stack Overflow. https://stackoverflow.com/q/64274976

4.Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. *International Journal of Information Management, 35*(2), 137-144. https://doi.org/10.1016/j.ijinfomgt.2014.10.007

5. Ai, J.-W., Zhang, Y., Zhang, H.-C., Xu, T., & Zhang, W.-H. (2020). Era of molecular diagnosis for pathogen identification of unexplained pneumonia, lessons to be learned. *Emerging Microbes & Infections, 9*(1), 597-600. https://doi.org/10.1080/22221751.2020.1738905

6. Li, W. (2011). On parameters of the human genome. *Journal of Theoretical Biology, 288*, 92-104. https://doi.org/10.1016/j.jtbi.2011.07.021

7. MongoDB. Comparison Query Operators — MongoDB Manual. Www.mongodb.com. https://www.mongodb.com/docs/manual/reference/operator/query-comparison/

8. Seguin, K.. About This Book. https://www.openmymind.net/mongodb.pdf