**MongoDB – Justification and Functional Documentation**

We chose MongoDB as the database for our project as MongoDB is a NoSQL database which stores the data in form of key-value pairs. It is an open source, document database which provides high performance and scalability along with data modelling and data management of huge sets of data in an enterprise application.

MongoDB also provides the feature of auto-scaling. It is a cross platform database and can be installed across different platforms like Windows, Linux etc. MongoDB’s document data model maps naturally to objects in application code, making it simple for developers to learn and use. Documents give you the ability to represent hierarchical relationships to store arrays and other more complex structures easily.

**Advantages of MongoDB**

1. MongoDB is a very easy to install and setup
2. MongoDB is a schema-less database. Developer code defines the schema.
3. As MongoDB supports document-based data model, it stores the data in a very rich way while being capable of holding arrays and other documents
4. MongoDB is ACID Compliant
5. MongoDB supports dynamic querying
6. Easily scalable database
7. Joins are not needed in MongoDB
8. Performance tuning is absolutely easy compared to any relational databases
9. No need of mapping the application objects to the data objects
10. Enables faster access of the data due to its nature of using the internal memory for storage
11. Being a NOSQL database, MongoDB is highly secure because no sql injection can be made
12. MongoDB can also be used as a file system, which helps in easier way of load balancing
13. MongoDB can be run as windows service as well
14. Good amount of documentation is available for easy understanding
15. MongoDB does not require a VM to be run
16. MongoDB supports horizontal scaling through Sharding, distributing data across several machines and facilitating high throughput operations with large sets of data.
17. Most importantly, it is very compatible with PowerBI

**Non-functional Requirements of MongoDB**

1. Performance – MongoDB uses a locking system to ensure data set consistency. If certain operations are long-running or a queue forms, performance will degrade as requests and operations wait for the lock.
2. Scalability - MongoDB supports horizontal scaling through Sharding, distributing data across several machines and facilitating high throughput operations with large sets of data.
3. Availability - MongoDB automatically maintains replica sets, multiple copies of data that are distributed across servers, racks and data centers. Replica sets help prevent database downtime using native replication and automatic failover.
4. Reliability - MongoDB has replica sets, which give hardware failover capabilities. Users can take backups on a regular basis, which gives them a recovery interval, and use sharding for some modicum of redundancy, especially when combined with replica sets.
5. Recoverability – Multiple data recovery strategies are available in MongoDB

* Backup Strategy #1: mongodump
* Backup Strategy #2: Copying the Underlying Files
* Backup Strategy #3: MongoDB Management Service (MMS)

1. Security - Storage engine, the primary component of MongoDB, is responsible for managing data. MongoDB provides a variety of storage engines, allowing you to choose one most suited to application. The Journal is a log that helps the database recover in the event of a hard shutdown. There are several configurable options that allows the journal to strike a balance between performance and reliability that works for particular use case.
2. Environmental - MongoDB is a cross-platform database
3. Data Integrity – MongoDB uses multi-document ACID transaction support. Multi-document transactions allow an application developer to write code that will start a transaction, modify multiple documents, and then commit that transaction. The commit will try and do all those changes unless it finds a problem. If an error occurs, the data in the database will remain unchanged.

**Functional Requirements of MongoDB**

**Downloading and installing MongoDB:**

MongoDB can be downloaded from official website. Refer to below link <https://www.mongodb.com/download-center/community>

**Installing BI connector for MongoDB:**

* Download the BI Connector from the [MongoDB Download Center](https://www.mongodb.com/download-center/bi-connector).
* Run the downloaded .msi file.
* Follow the wizard instructions to install the files. The binaries install into a bin directory inside the installation directory.
* Launch the BI connector using 3 ways:

1. Start mongosqld from the command Line

2. Mongosqld with a Configuration file

3. Install mongosqld as a System Device

**Installing ODBC connector for the system:**

* Download ODBC connector from the MongoDB website.
* In order to configure ODBC driver DSN needs to be setup.

**Steps for creating DSN :**

* Choose the program version (64-bit or 32-bit) which is appropriate for your system and ODBC driver version.
* Select the System DSN tab.
* Click the Add button.
* Select either the MongoDB ODBC 1.1.0 ANSI Driver or the MongoDB ODBC 1.1.0 Unicode Driver, then click OK.
* Fill the necessary form fields.
* If authentication is enabled then fill relevant fields.
* If TLS/SSL is enabled, fill in the TLS/SSL form fields.
* Click the Test button to test the ODBC connection.

**Prerequisites for installing Power BI on Desktop:**

* Windows 7 / Windows Server 2008 R2, or later
* .NET 4.5
* Internet Explorer 9 or later
* Memory (RAM): At least 1 GB available, 1.5 GB or more recommended.
* Display: At least 1440x900 or 1600x900 (16:9) recommended. Lower resolutions such as 1024x768 or 1280x800 are not recommended.
* Windows Display settings: If display settings are set to change the size of text, apps, and other items to more than 100%, users may not be able to see certain dialogs that must be closed or responded to in order to proceed using Power BI Desktop. If users encounter this issue, check Display settings by going to Settings > System > Display in Windows, and use the slider to return display settings to 100%.
* CPU: 1 gigahertz (GHz) or faster x86- or x64-bit processor recommended.

**Installing Power BI:**

* Download Power BI desktop from Microsoft website or from Microsoft App Store.
* Install as an app from the Microsoft Store.

**Steps Involved during MongoDB database implementation:**

**Analyzing the dataset**

Dataset needs to be analyzed, understand the metadata and how that data could be useful to business. Metadata documentation needs to be performed.

**Data Cleaning**

Preliminary analysis on the data was performed and prepared the data for extracting useful insights. Data had to be cleaned, formatted so that it could be used for analysis. Data Cleaning was performed on Jupyter Notebook using Python.

*a) Identified null values*: There were many missing values in the dataset. Rows with null values were removed from the analysis.

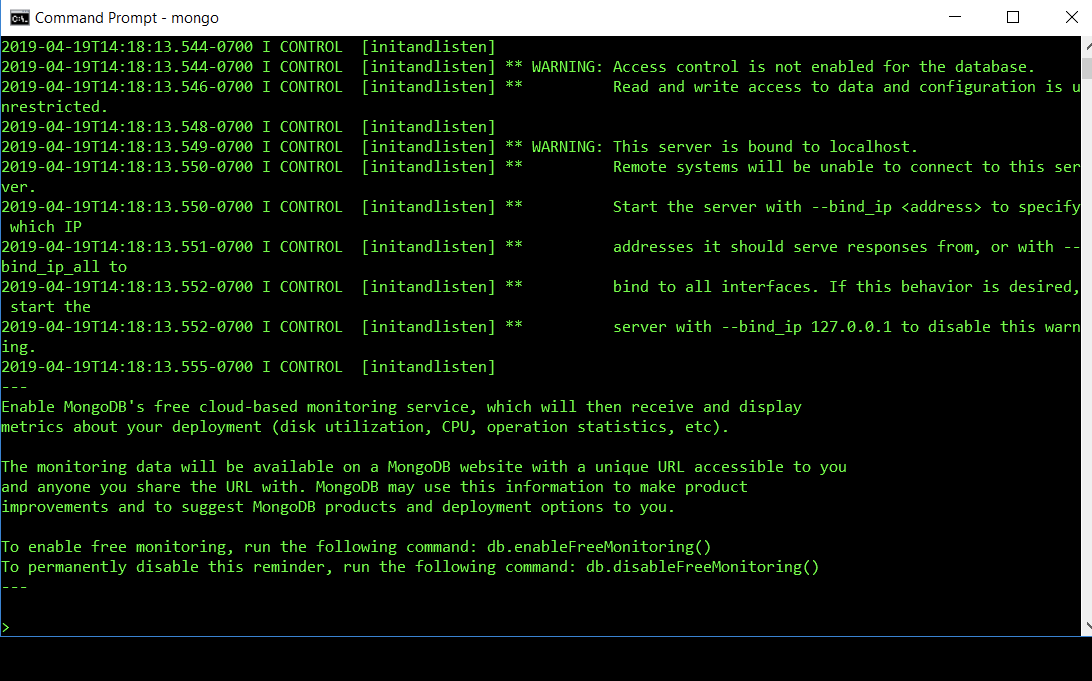
*b) Date formats*: Data was formatted and extracted in CSV and JSON formats so that it could be used for analysis.

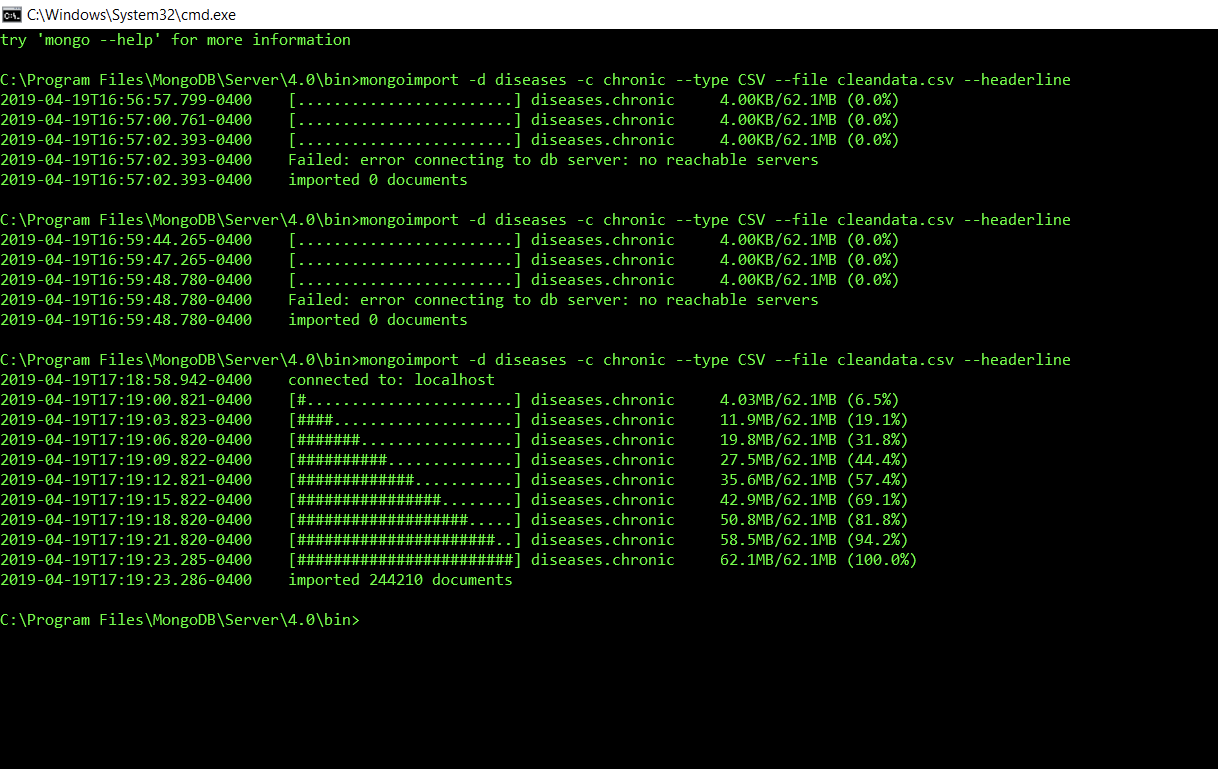
**Importing the cleaned data into MongoDB:**

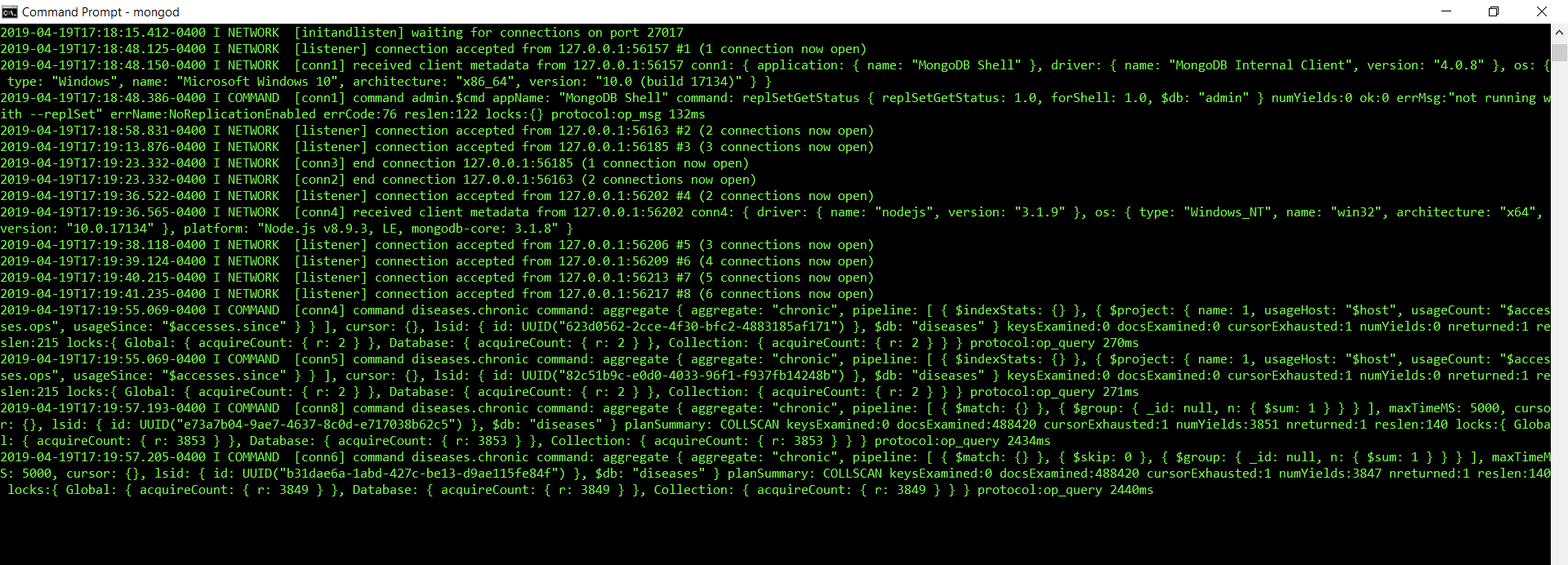
The cleaned data in CSV format was imported into MongoDB using the below script -

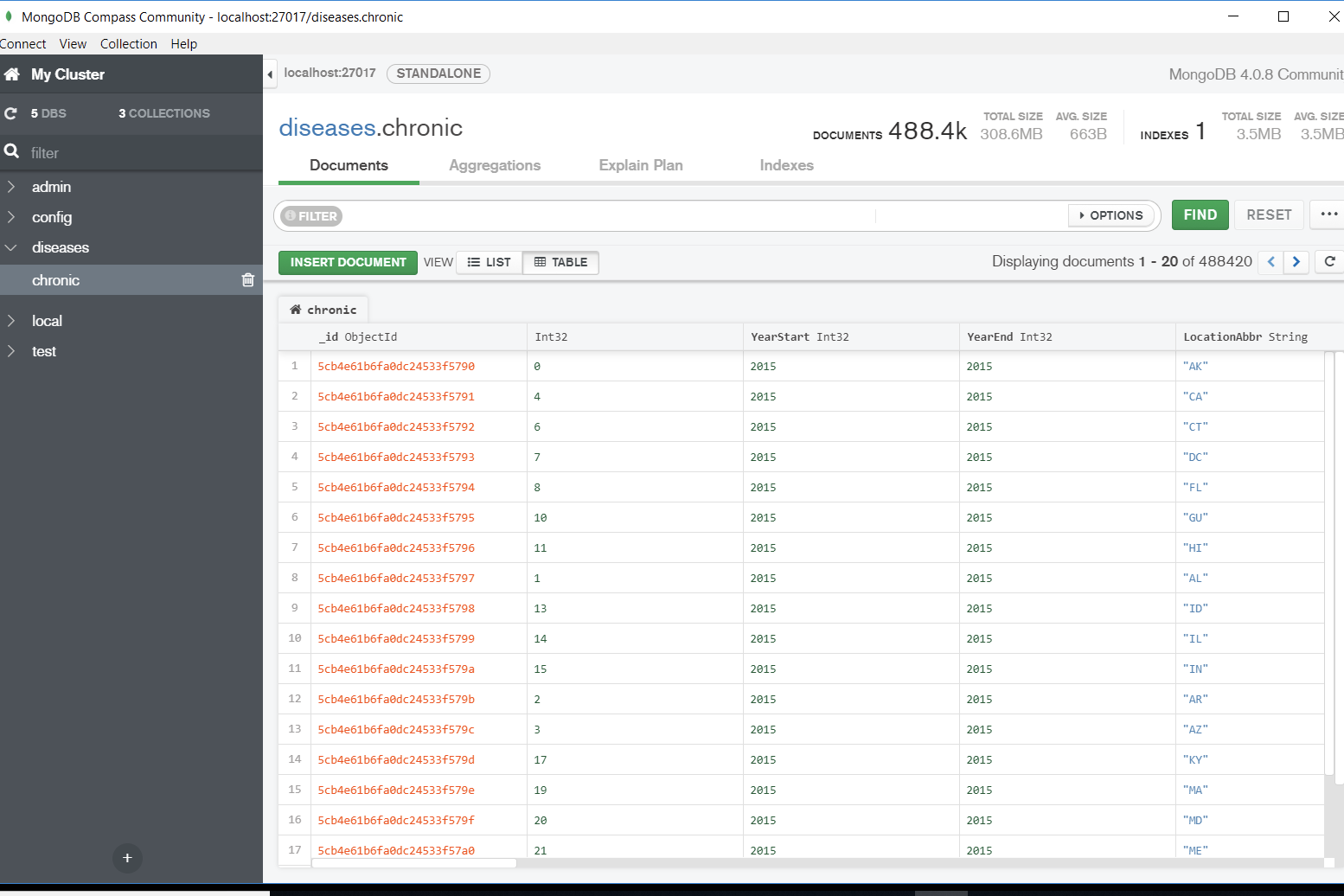
C:\Program Files\MongoDB\Server\4.0\bin>mongo -d diseases -c chronic --type CSV --file cleandata.csv --headerline

Please refer below snapshots of below screens for importing process -









**Connecting Power BI to MongoDB:**

Start the mongosqld process by executing below script -

C:\Program Files\MongoDB\Connector for BI\2.1\bin\mongosqld.exe

Further, create a Data Source Name (DSN) and then connect it with Power BI.