Gabriela Masak

D597 – Data Management

Task 2: NoSQL Databases

9/5/2024

Design Document: Scenario 1

**Part 1: Design Document**

Healthcare is constantly evolving, with new research and findings leading to advances in techniques, diagnoses, and treatments. Many of these improvements are led by data findings. As such, many healthcare organizations are focusing their efforts on building database management systems (DBMS) that can efficiently handle the scalability, flexibility, and efficiency necessary, while also maintaining the ethical standards of security. These systems can be defined by their data model, entities, attributes, data organization, scalability, security, and integration capabilities. Each catering to the unique business problem they collectively solve.

For HealthFit Innovations, this business problem is related to its exponential growth as the company attempts to launch a new platform, HealthTrack, and analyze the data to determine the best approach for marketing and development. Its users are constantly uploading structured and unstructured data related to heart rate, sleep patterns, activity levels, and blood glucose levels from wearable technology like smart watches and medical devices. The volume of real-time, diverse, and dynamic data is rapidly increasing in a manner that its current relational DBMS cannot handle. This can be solved by implementing a new non-relational DBMS with the ability to scale while integrating with existing systems, removing issues like performance bottlenecks, data silos, and analysis challenges.

There are several different types of non-relational databases. The most popular are key-value, document-based, column-family, graph, object, multi-model, and grid and cloud databases (Keita, 2022). The database solution that would most effectively function is a cloud document-based NoSQL database, like MongoDB Atlas, where JSON files are stored as documents on an online server featuring a non-relational database design. This is optimal as HealthFit Innovations’ data is multiuser, discipline-specific, and features various data types with varying degrees of structure uploaded in real time in JSON files. Not to mention, due to the company’s need to rapidly migrate from its current program, the quick abilities of a cloud-based analytical database would furnish this migration in minutes rather than weeks or months, marking the event with little disruption. Cloud databases can also be scaled easily and can keep up with rapid user growth. Devices such as smart watches and medical devices need to be connected to the internet for functionality, falling under the definition of Internet of Things. This connection allows for data to be sent via wireless networks in real time, which means that routing this data to the cloud for both storage and analysis would be highly feasible. Data would be uploaded as documents, grouped into collections for organization. Business users would also be able to run applications and queries on the cloud and business clients would be able to view and access their personal data and derived insights.

HealthFit Innovation’s new database would be comprised of a data warehouse and an analytical processing front end. Data sent from smart watches and medical devices will be stored as documents in the data warehouse as JSON files. A set of tools in the analytical processing front end, called online analytical processing, provide the environment where the data can be retrieved from the data warehouse for processing and modeling. HealthFit Innovations features data comprised of details about smartwatches and client health information. Smartwatch information includes brand name, model name, color, selling price, original price, display, rating, strap material, and average battery life. Client health information consists of patient id, name, date of birth, gender, medical conditions, medications, allergies, and date of last appointment.

These documents are organized in two collections, smartwatches and health data. This organization allows for data to be queried for analysis that can support and manage business functions, like determining whether partnering with specific brand of smartwatch that is exceedingly popular is a possibility. The analysis could show whether a certain demographic is more inclined to use smartwatches for marketing purposes. Lastly, queries can show if there are common shared medical conditions, allowing researchers to focus on developing new methods and applications that allow better health management geared towards their specific condition.

MongoDB can address scalability concerns by use of replication and sharding. Replication is the process of copying data across multiple servers to ensure high availability and redundancy. This means that if one server fails, another can take over, minimizing downtime and potential data loss. Replication also allows for load balancing during read operations, as data can be read from multiple servers, enhancing performance and reliability. Sharding is the method of distributing data across multiple servers to balance the load and improve performance by dividing large datasets into smaller, more manageable pieces called shards. It enables horizontal scaling, where more servers can be added to handle increased load (Prakash, 2021). By leveraging MongoDB’s replication and sharding capabilities, HealthFit Innovations can efficiently manage the vast and diverse data generated in real-time from various sources, ensuring high availability, scalability, and performance.

Due to the ethical and legal requirements surrounding sensitive health data, several privacy and security measures can be enacted. Users and their abilities to access and manipulate data can be controlled by implementing authentication and authorization methods. In MongoDB, data can be encrypted, users can be authenticated and authorized, with their activity tracked to ensure that access and modifications are acceptable (NoSQL, 2024). For HealthFit Innovations, this means implementing login credentials for clients, healthcare individuals, and data professionals with different levels of authorization.

The rapid evolution of healthcare necessitates robust and scalable database management systems to handle the increasing volume and variety of data. HealthFit Innovations faces a significant challenge due to its exponential growth and the diverse data generated from wearable devices and medical records. Transitioning to a cloud-based document-oriented NoSQL database like MongoDB Atlas can effectively address these challenges by providing scalability, flexibility, and real-time data processing capabilities. This solution not only ensures high availability and performance through replication and sharding but also supports the integration of various data types and sources. By leveraging MongoDB’s advanced features, HealthFit Innovations can enhance its data management and analytics capabilities, ultimately leading to improved healthcare outcomes and more personalized health insights for its users.

**Part 2: Implementation in the WGU Virtual Lab environment**

1. Write script to create a database instance named “D597 Task 2” using the appropriate query language, based on your design in Part 1. Provide a screenshot showing the script and the database instance in the platform.

A screenshot of a computer

Description automatically generated

Figure 1: Screenshot of Script and D597 Database Instance

2. Write script to insert or map the data records from the chosen scenario JSON files into the database instance. Provide a screenshot showing the script and the data correctly inserted or mapped into the database.

A screenshot of a computer

Description automatically generated

Figure 2: Screenshot Showing Documents in Smartwatches Collection

A screenshot of a computer

Description automatically generated

Figure 3: Screenshot Showing Documents in HealthData Collection

3. Write script for three queries to retrieve specific information from the database that will help to solve the identified business problem. Provide a screenshot showing the script for each query and each query successfully executed.

Query 1: Most Popular Brands of Smartwatches

db.Smartwatches.find({}, {“Brand Name”: 1}).sort({“Reviews” : 1}).limit(3);

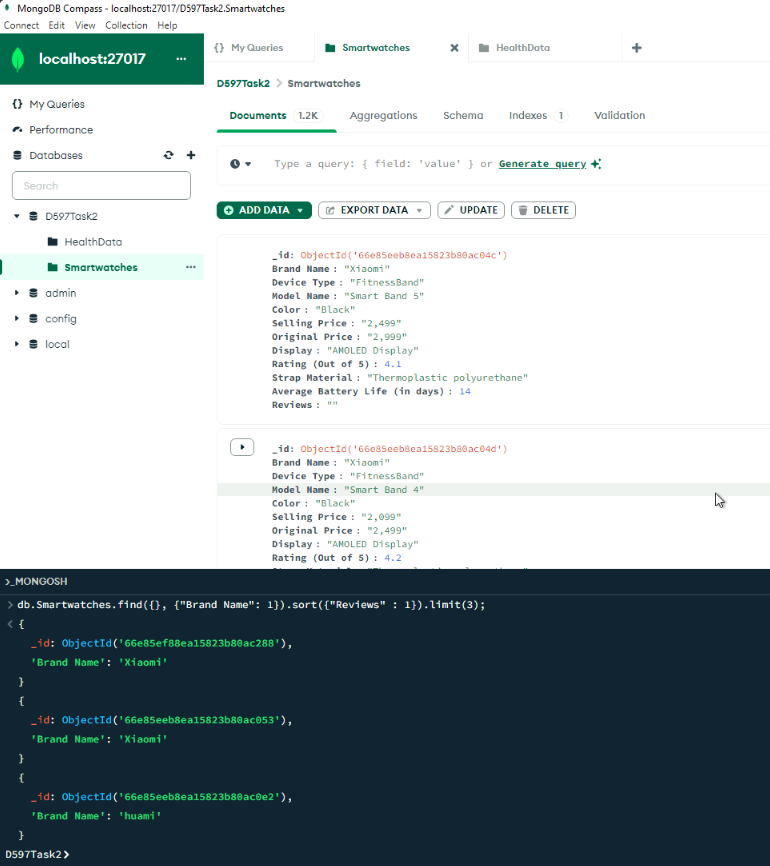


Figure 4: Screenshot of First Query and Results

Query 2: Determine Demographics

db.Smartwatches.find({}, {“gender”: 1}).count({ “gender” : “M” });

db.Smartwatches.find({}, {“gender”: 1}).count({ “gender” : “F” });

A screen shot of a computer code

Description automatically generated

Figure 5: Screenshot of Second Set of Queries and Results

Query 3: Common Shared Medical Conditions

db.HealthData.aggregate([

{ $unwind: "$medical\_conditions" },

{ $group: { \_id: "$medical\_conditions", count: { $sum: 1 } } },

{ $sort: { count: -1 } },

{ $limit: 3 }

])

A screen shot of a computer program

Description automatically generated

Figure 6: Screenshot of Third Query and Results

4. Apply optimization techniques to improve the run time of your queries from part D3, providing output results via a screenshot.

Optimized Query 1: Most Popular Brands of Smartwatches

db.Smartwatches.find({}, {“Brand Name”: 1}).sort({“Reviews” : 1}).limit(3).explain("executionStats");



Figure 7: Screenshot of Results of Query 1 Optimization Attempt

Query 2: Determine Demographics

db.Smartwatches.explain("executionStats").aggregate([ { $group: { \_id: "$gender", count: { $sum: 1 } } }]);

**A screen shot of a computer code

Description automatically generated**

Figure 8: Screenshot of Results of Second Query Optimization Attempt

Query 3: Common Shared Medical Conditions

db.HealthData.explain("executionStats").aggregate([

{ $unwind: "$medical\_conditions" },

{ $group: { \_id: "$medical\_conditions", count: { $sum: 1 } } },

{ $sort: { count: -1 } },

{ $limit: 3 }

]);

A screen shot of a computer code

Description automatically generated

Figure 9: Screenshot of Results of Third Query Optimization Attempt

**Part 3: Presentation**

The audiovisual presentation of the NoSQL database, design, queries, and implementation can be found at the following hyperlink:

<https://wgu.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=fe96a13a-e4ab-491b-83f5-b1ee01011487>

**Sources Cited**

1. Keita, Z. (2022, June). *NoSQL databases - types of nosql databases and how to use*

*them*. www.datacamp.com; DataCamp, Inc.

https://www.datacamp.com/blog/nosql-databases-what-every-data-scientist-needs-

to know/

1. *NoSQL Database Security*. (2024). MongoDB White Paper; MongoDB.

https://www.mongodb.com/resources/products/capabilities/nosql-database-

security?msockid=0c5f727cc54367960929669dc4b26657

1. Prakash, Ashutosh. (2021, October 16). *MongoDB Replication and Sharding*.

GeeksforGeeks; GeeksforGeeks. https://www.geeksforgeeks.org/mongodb-

replication-and-sharding/#