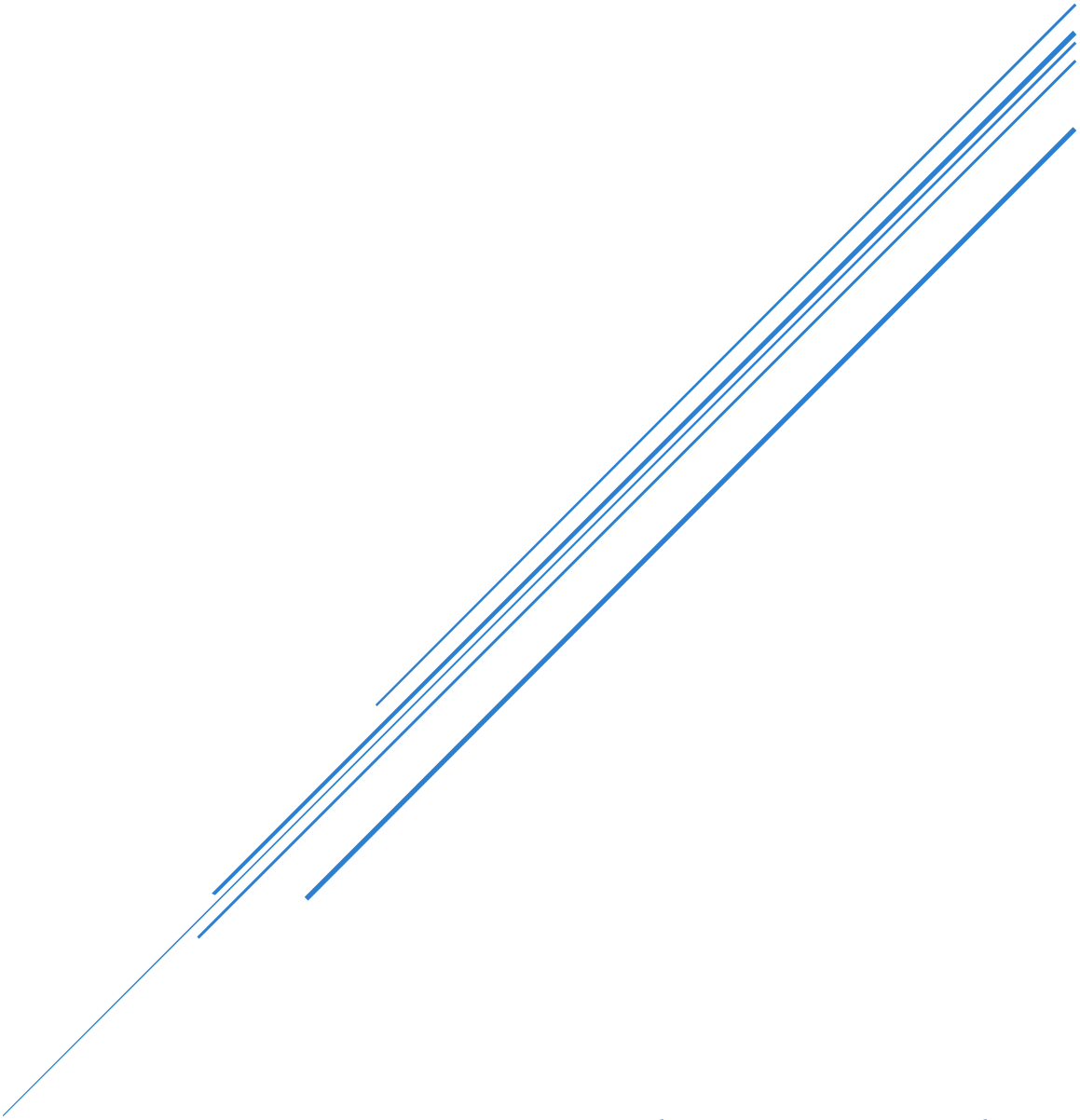
HealthFit Database Design

Design, Implementation, and Optimization for Analytics



Author: Joanna Ronchi

Date: 05/24/2025

Table of Contents

Part 1: HealthFit Design Document

A1. Description of the business problem

Supporting the Business Mission

HealthFit is a health-focused business selling a data solution that enables customers to track their health data. The company provides data-driven recommendations for customers and their healthcare providers to help them improve health.

Top Business Needs

The business needs to upgrade its current database solution to address scalability, improve performance, and integrate data from heterogeneous sources. The top business problems include data silos, resulting in delaying health recommendations, and reducing the value of Health Fit's HealthTrack platform. The business also suffers from scalability bottlenecks due to rapid growth. The bottlenecks slow down the processing of data and hinder business progress. Health Track's real-time data processing issues, caused by large volumes of incoming data, create issues during the processing stage of the data lifecycle, resulting in processing delays or incorrect outputs.

Top Solution Considerations

To solve the problem, the business needs to increase the capacity of computing resources. HealthFit needs the new system to integrate complex facts – including biometric, clinical, and lifestyle data - for accurate analysis and insights. The business needs a solution to integrate and unify multiple source data, handle nested information, and unify patient information from various sources into a single document. HealthFit Innovations needs to connect the siloed and disconnected data to provide value for customers and ensure the accuracy of crucial health insights.

A2. Justification for NoSQL database solution for the business problem

Schema Flexibility

A NoSQL database will provide schema flexibility for storing varying data from multiple sources and categories. Since the company's health data varies dramatically between heart rate, blood glucose, step count, sleep and activity patterns, blood oxygen saturation, body temperature, or GPS data (for measuring distance traveled), the relational data model cannot effectively accommodate HealthFit Innovations' business needs.

Horizontal Scalability

No SQL solutions will allow the system to scale out using sharding, enabling scalability as the volume of data grows. NoSQL databases fit nicely into dynamically changing, semi-structured data from wearable devices because they do not sacrifice performance as they distribute the load between several nodes (Dey, 2020).

Supporting Real-Time Data Performance

The business collects real-time data from wearable devices, needs to support real-time data monitoring, and requires high write throughput.

Data Structure Compatibility

The NoSQL database will support document-based data more effectively than relational models by supporting nested data and arrays. The NoSQL data model directly represents hierarchical relationships and manages complex structures by including nested arrays and objects in the data

documents. The structure eliminates the need for complex tables and designs in relational databases, improving performance and easier access to data with simpler queries (Blefqih, 2024).

A3. Proposed NoSQL database type to solve the identified business problem

MongoDB will support storing and analyzing semi-structured customer health data, such as JSON documents from wearable devices and applications. The database will give the business flexibility and support inputs from various users, including healthcare providers and health-data reporting systems. MongoDB will support real-time data reporting and provide performance optimization capability, support indexing, aggregation functionality, and real-time analytics (Wolniak, 2023).

A4. How the business data will be used within the database solution

The data from HealthTrack will be used for real-time health-data monitoring and analysis to quickly identify and provide critical health alert data, such as heart rate measurements and alerts to customers. The database's functionality will enable the generation of health insights based on past data and identify potential health threats using predictive analytics. The solution will use data to analyze early warning signs of illness and allow users to make health-related decisions. The system will enable dashboard generation for various users, including patients and their healthcare providers, to manage and improve patient health, track improvement progress, and generate customized health tips.

B. How the proposed database design addresses scalability concerns

NoSQL Database

Rigid, relational schema designs are unsuitable for the HealthTrack business, primarily due to inflexibility. HealthFit Innovations solution needs a scalable database that supports millions of users uploading real-time, continuous, and diverse data related to varying health metrics. A document-oriented model offered by MongoDB will accommodate the scalability and performance and offer flexibility for the business to handle the massive amounts of data coming into the system.

Sharding and Horizontal Scaling

MongoDB allows for the distribution of data across servers, which is known as sharding. The nodes distribute data based on parameters such as `user_id` or `device_id`. The database will allow Heath Fit innovations to add more users and increase without sacrificing performance by adding servers to accommodate the growth. The additional servers can handle specified portions of read/write traffic, enabling throughput and preventing failure due to traffic increases. Health monitoring data from mobile devices is high-velocity and requires strong, scalable, and flexible systems. NoSQL databases support mobile device health data and efficiently scale (He, 2021).

Real-Time Monitoring

NoSQL databases allow for high-speed writes of health data from wearable devices and can effectively store time-series data. Healthcare IoT devices handle data better than traditional relational models (Dey, 2020).

Flexibility

The variability of health data requires flexible data schemas to monitor diverse and evolving data structures such as heart rate metrics, blood glucose levels, medication compliance logs, or sleep

and activity data. Each type of health data requires a customized model, and the NoSQL database will enable the business to structure this data for accurate analysis (Alshammari, 2019). MongoDB offers a schema-less design and adds data without typical relational model complexities. The flexibility of the MongoDB database will enable agile development and growth of the data from various users as the business expands its customer base.

Data Durability

NoSQL models provide data redundancy by storing data in multiple copies across servers and safeguarding the data against loss. In case of failure of a single node, another server assumes control over the data, minimizing downtime. The fault tolerance capability supports data availability, which is particularly important in environments that handle critical health data.

Real-Time Analysis

MongoDB offers a unique aggregation pipeline capability to process, transform, and analyze data. HealthFit data can use this feature to quickly and efficiently generate insights from real-time incoming patient/customer data and enables quick generation of trigger alerts such as abnormal heart rate (Zhang, 2019). Aggregation pipelines are efficient, flexible, and important, especially for health monitoring and IoT in healthcare settings. The business can provide critical health insights, offer value to customers, and advertise the feature's availability for effective marketing campaigns.

C. Privacy and security measures

Compliance with HIPPA and GDPR

The U.S. Health Insurance Portability and Accountability Act (HIPPA) and the European Union's General Data Protection Regulation (GDPR) Act require businesses dealing with health information to implement guardrails to protect the safety and privacy of individual health information. Pseudonymization replaces data that identifies individuals with fields such as patientID or userID. Anonymization is another technique HealthFit should use to perform the analysis while protecting individual privacy. The data masking methods will enable safe data sharing and meet the legal obligations of a company dealing with highly sensitive health data (Wang, Zhang, 2020).

Security and Privacy of Health Data

HealthFit Innovations has legal obligations to protect the privacy and security of its users. The database must provide field-level security to protect patient names, social security numbers, and other sensitive data (Fernandez, 2013). HealthFit needs to protect the data at rest and in transit. MongoDB supports the industry-standard AES-256 and Transport Level Security, offering additional client-level encryption (MongoDB, n.d., p. 1).

Role-Based Access Control and Audit Logging

The business must follow the principle of least privilege to protect patient and user personal information based on necessity, especially in cloud-based health data (Gayanajake, 2017). Users and patients should be able to access all their data and assign permissions to their healthcare providers. Data analysts and scientists should only have access to anonymous and pseudonymized data sets. The business will implement audit logging to track and monitor who and when accessed the data and made changes. Configuring the database with RBAC (Role-Based Access Controls) is important. An important MongoDB feature that will enable data confidentiality, integrity, and

access controls is the system built-in security.authorization: enabled setting, which configures MongoDB to require users' authentication based on their roles, for example admin, read-only, or root access.

Backups

MongoDB will provide automated backups to prevent data loss, which is critical in healthcare environments. The database will redundantly and securely store data in several locations using snapshots and encryption. MongoDB enables regular, scheduled backups at convenient times. The database offers replicas where one node writes, and a secondary node reads the data, preventing loss and failure. MongoDB documentation has extensive information on replication functionality, assuring the business that the solution will provide real-time configured backups and increase the security of data (MongoDB, n.d., p. 2).

Specific MongoDB Features Strengthening Security

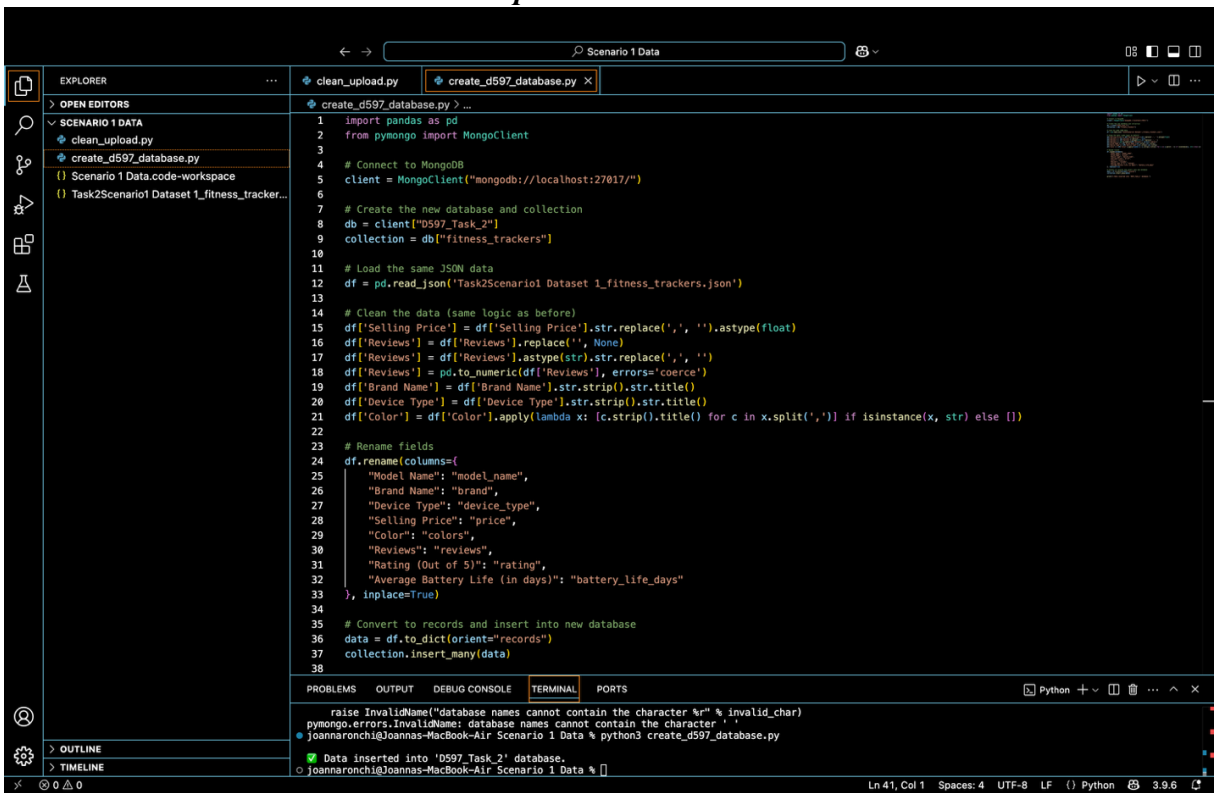
MongoDB comes with built-in security features. It is important to configure settings to prevent attacks and protect sensitive data. The recommended settings include:

- net.port—The feature changes the default listens to on position, reducing cyberattack vulnerabilities that scan MongoDB ports.
- net.bindIp—This feature limits access to the database to only specified IP addresses and prevents unauthorized access and misappropriated data use.
- javascript-enabled – set to disable or prevent Java script injection attacks. MongoDB will reduce the attack surface from untrusted queries or that are not sanitized inputs (MongoDB, n.d., p. 3)

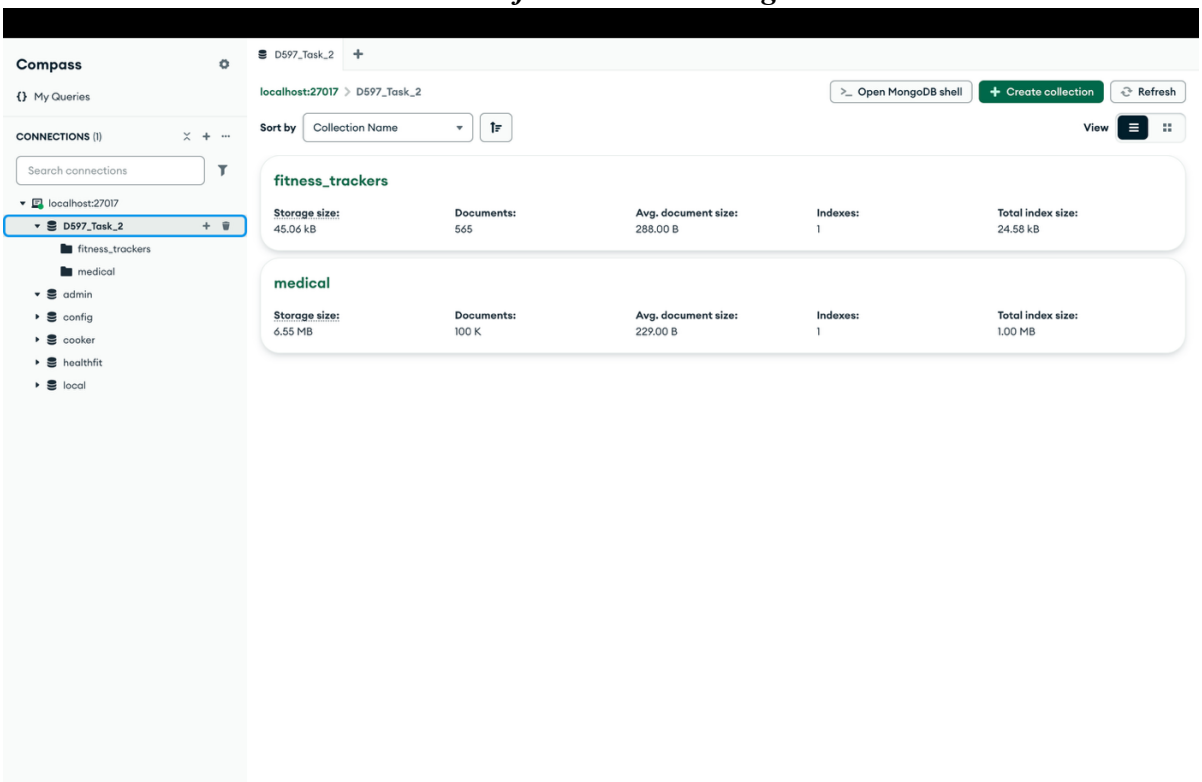
Part 2: Implementation

D1. Database instance screenshots (Python and JavaScript)

Database Script Created in Visual Studio:



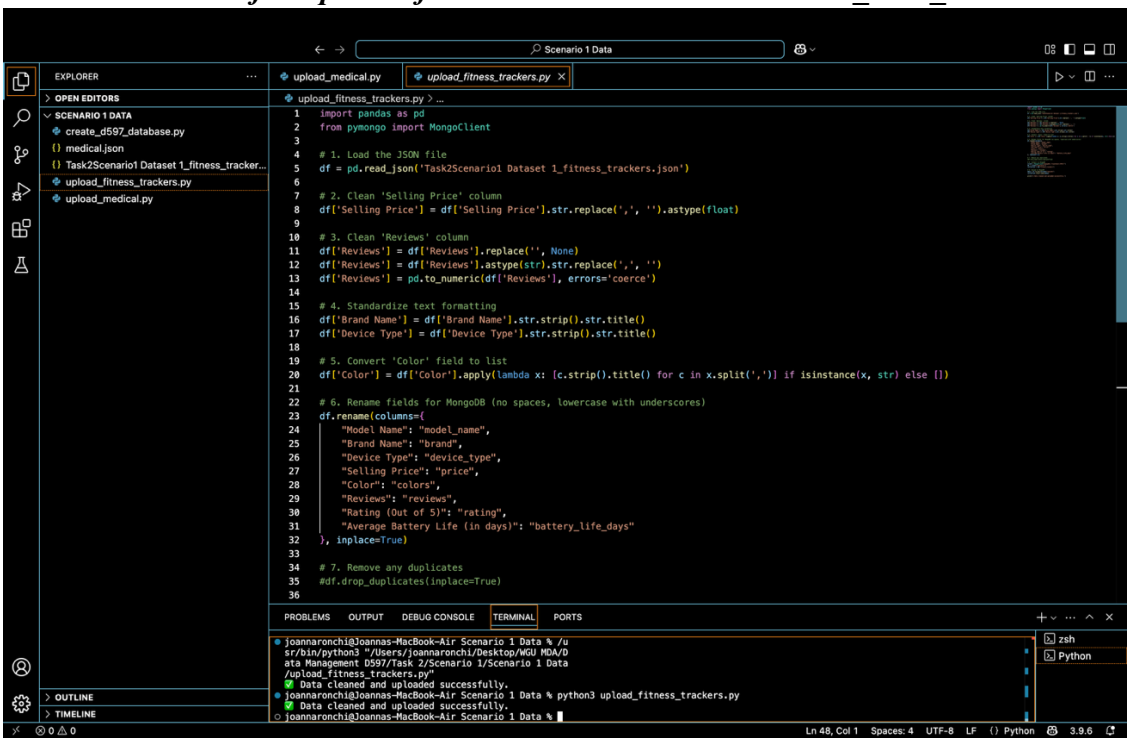
Screenshot of Database in MongoDB:



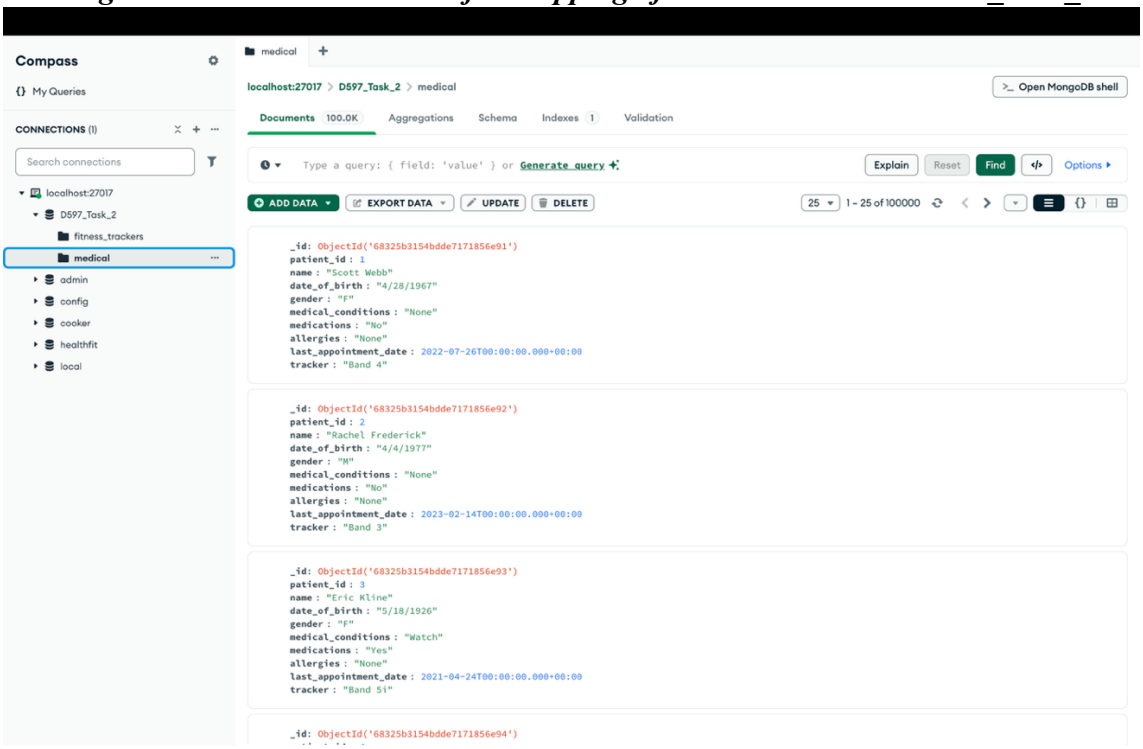
D2. Script to insert and map the data records into the database instance.

Successful Upload of Medical Data into D597_Task_2:

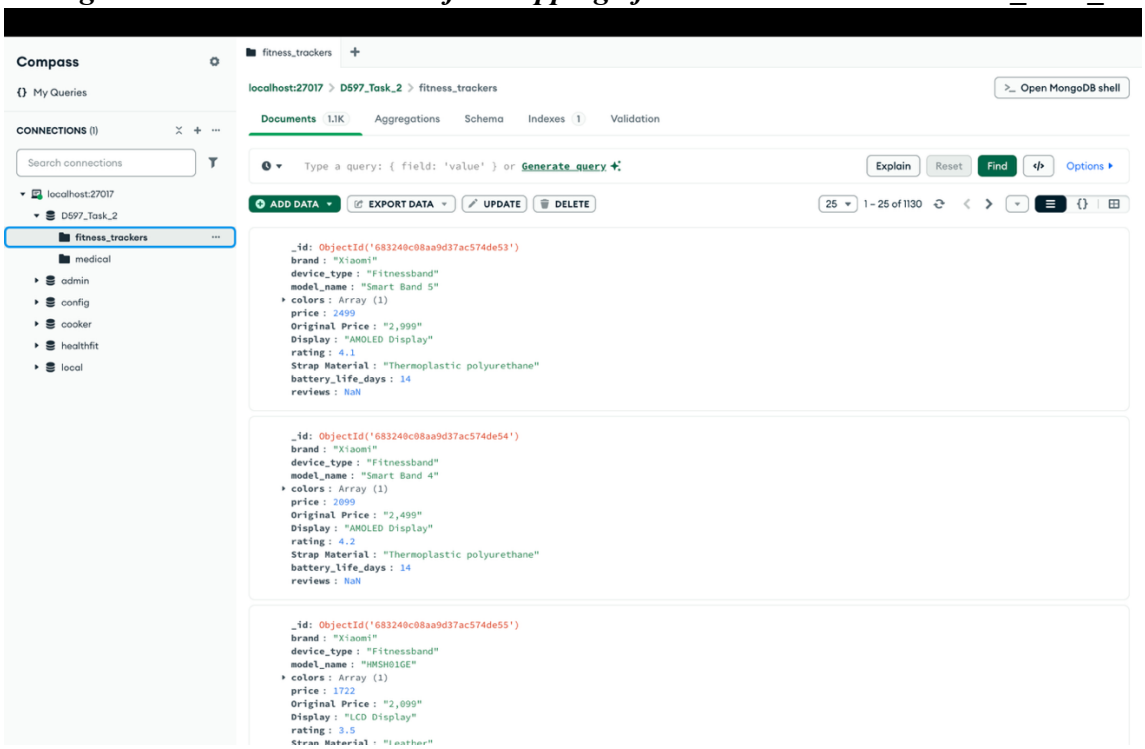
Successful Upload of Fitness Tracker Data into D597_Task_2:



MongoDB Screenshot – Successful Mapping of Medical Data into D597_Task_2



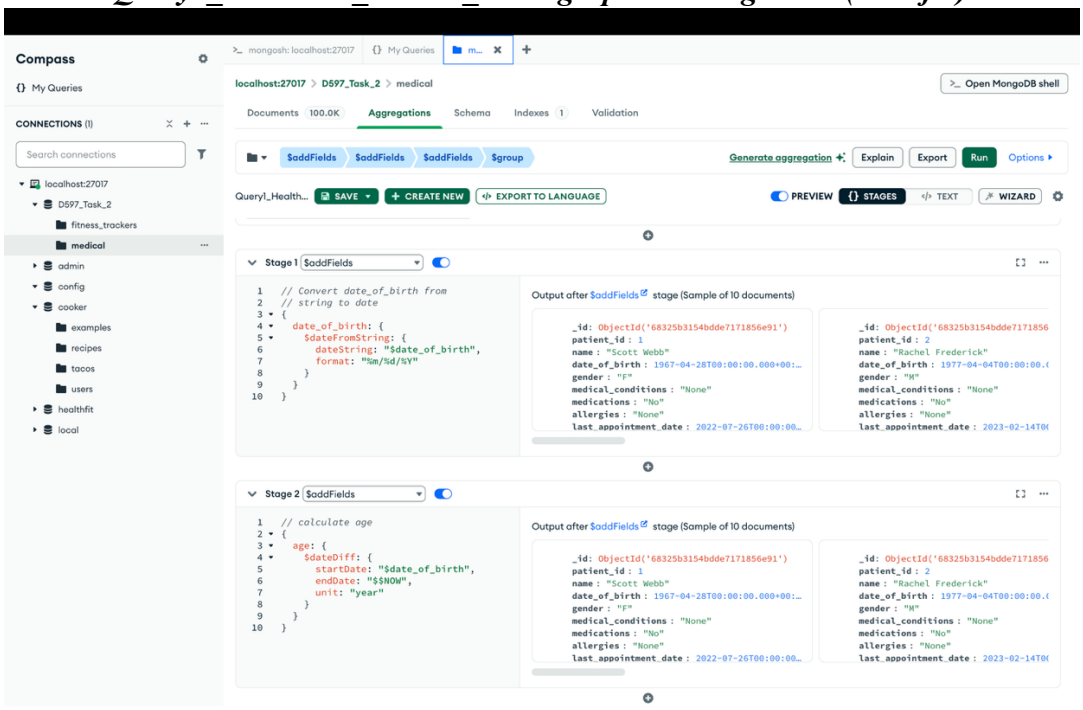
MongoDB Screenshot – Successful Mapping of Fitness Trackers into D597_Task_2



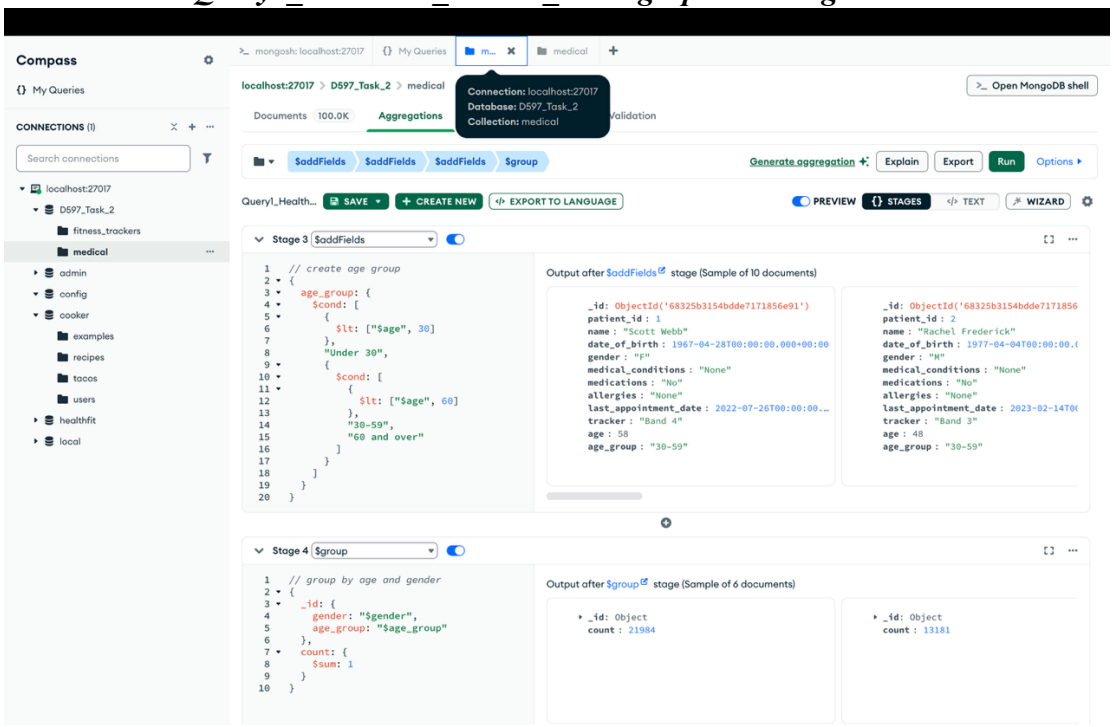
D3. Query Scripts

QUERY 1

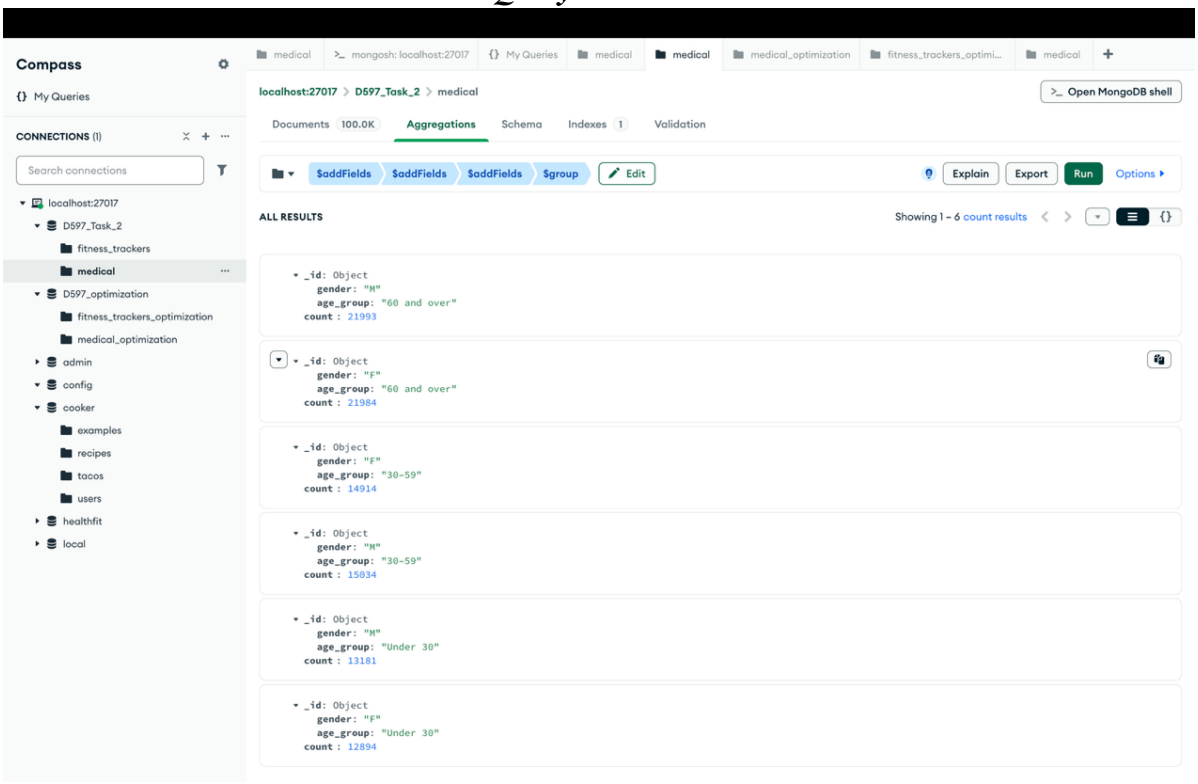
Query1_HealthFit_Patient_Demographics - Stage 1&2 (out of 4)



Query1_HealthFit_Patient_Demographics - Stage 3&4

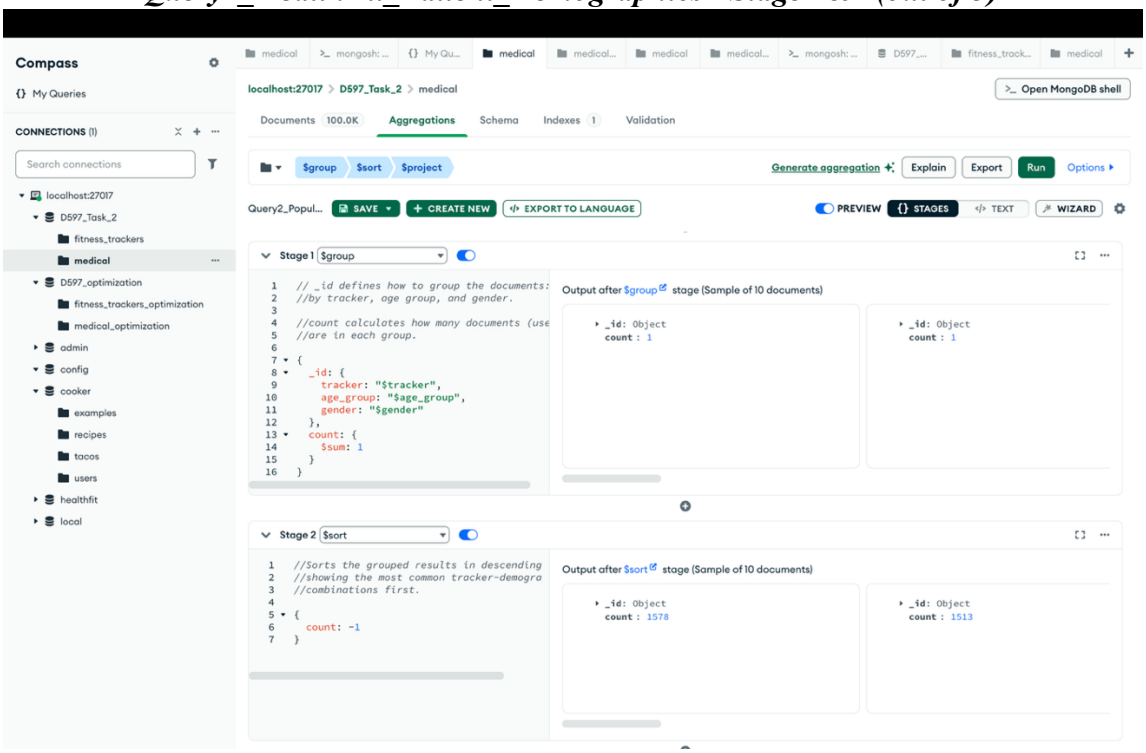


Query 1 Results

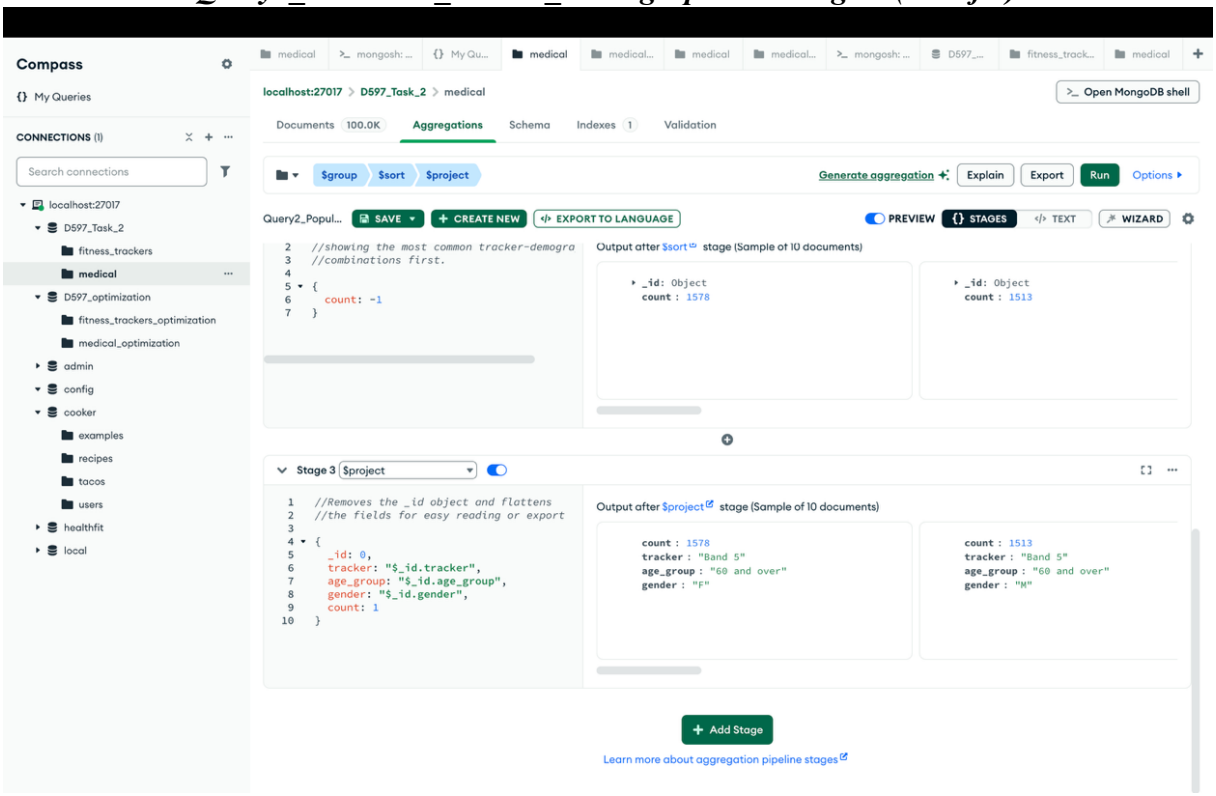


QUERY 2

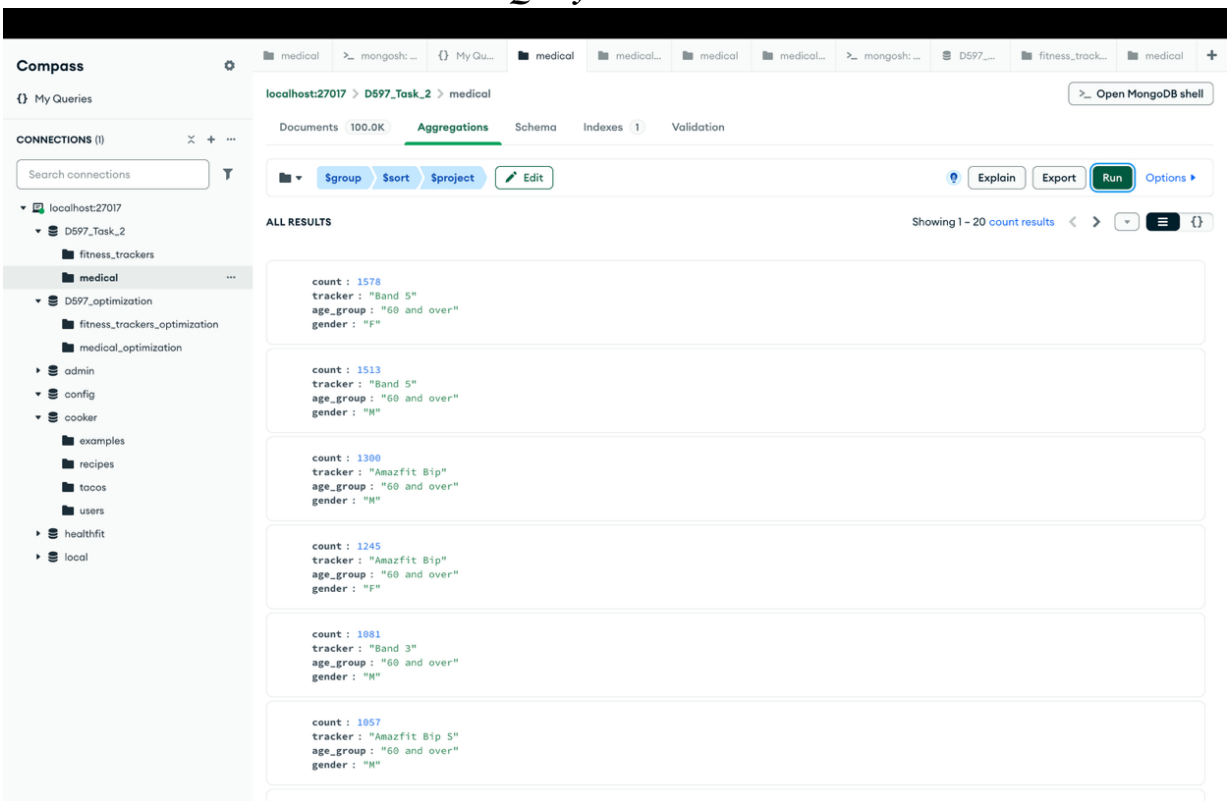
Query2_HealthFit_Patient_Demographics - Stage 1&2 (out of 3)



Query2_HealthFit_Patient_Demographics - Stage 3 (out of 3)

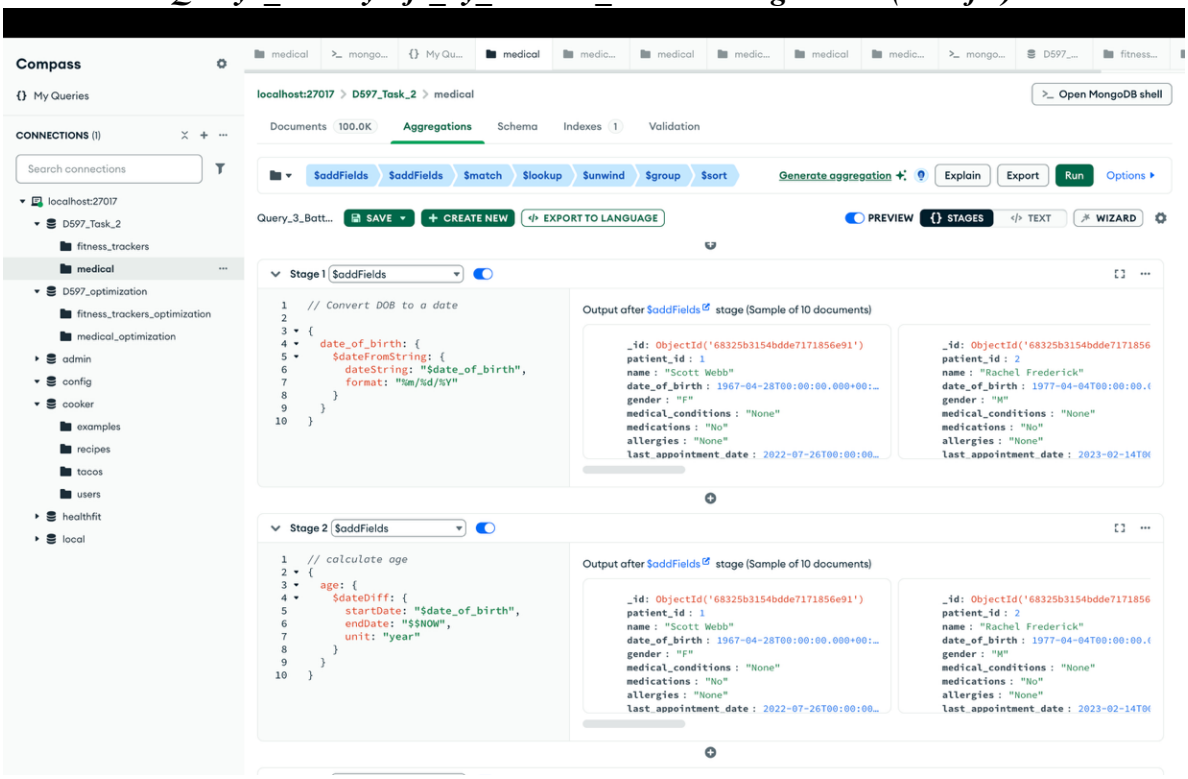


Query 2 Results

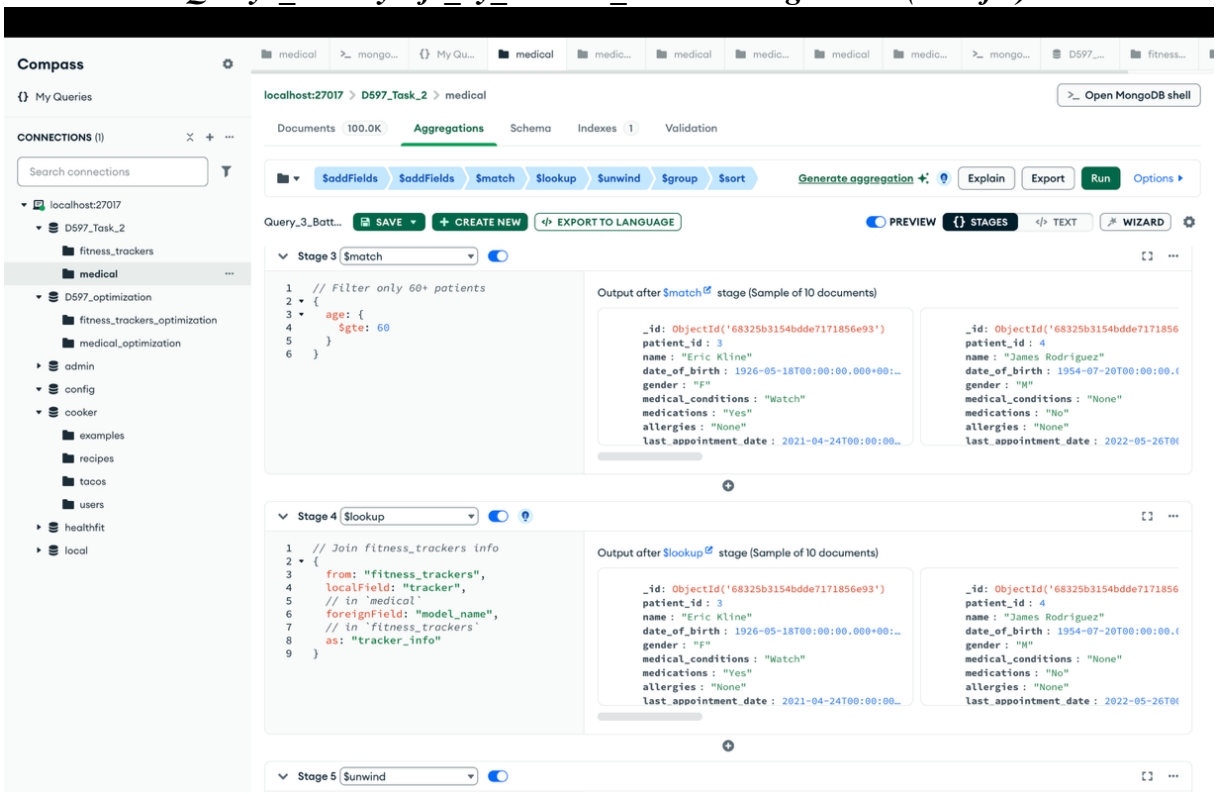


QUERY 3

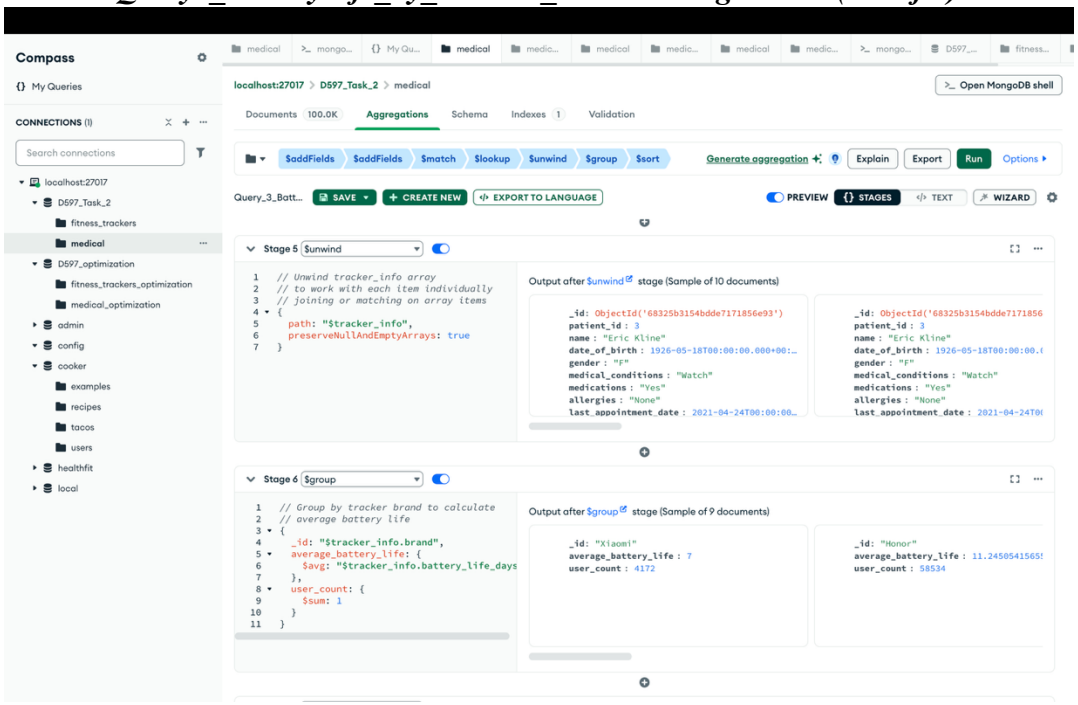
Query3_BatteryLife_by_Tracker_Brand – Stage 1 & 2 (out of 7)



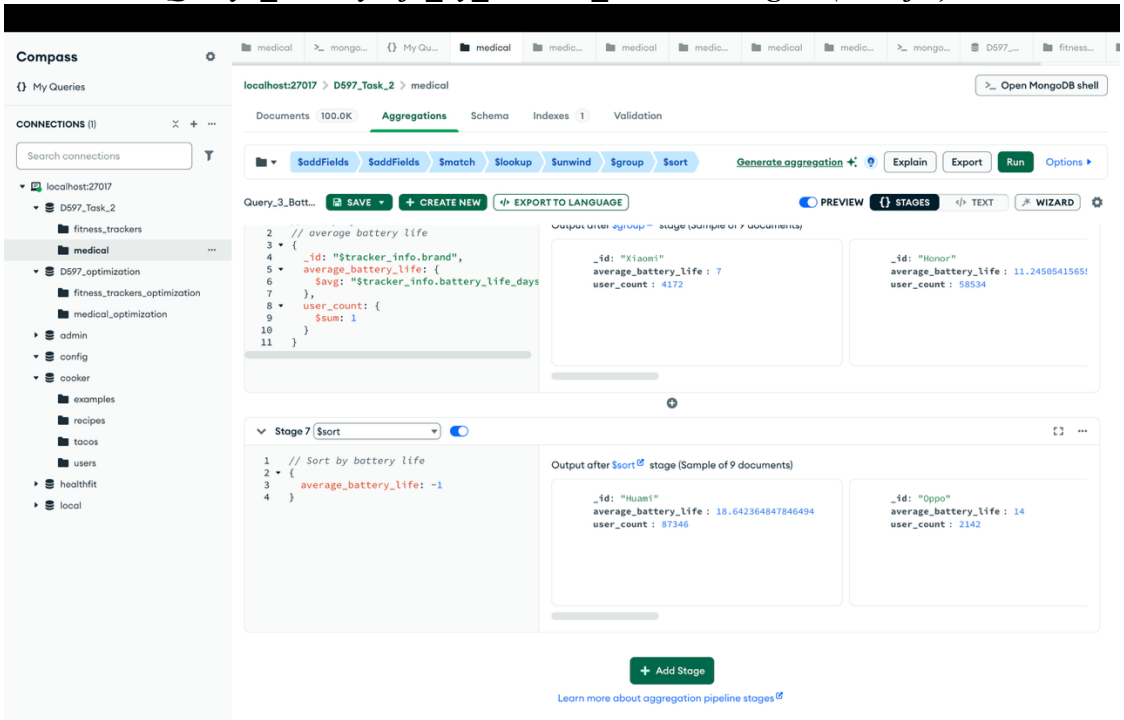
Query3_BatteryLife_by_Tracker_Brand – Stage 3 & 4 (out of 7)



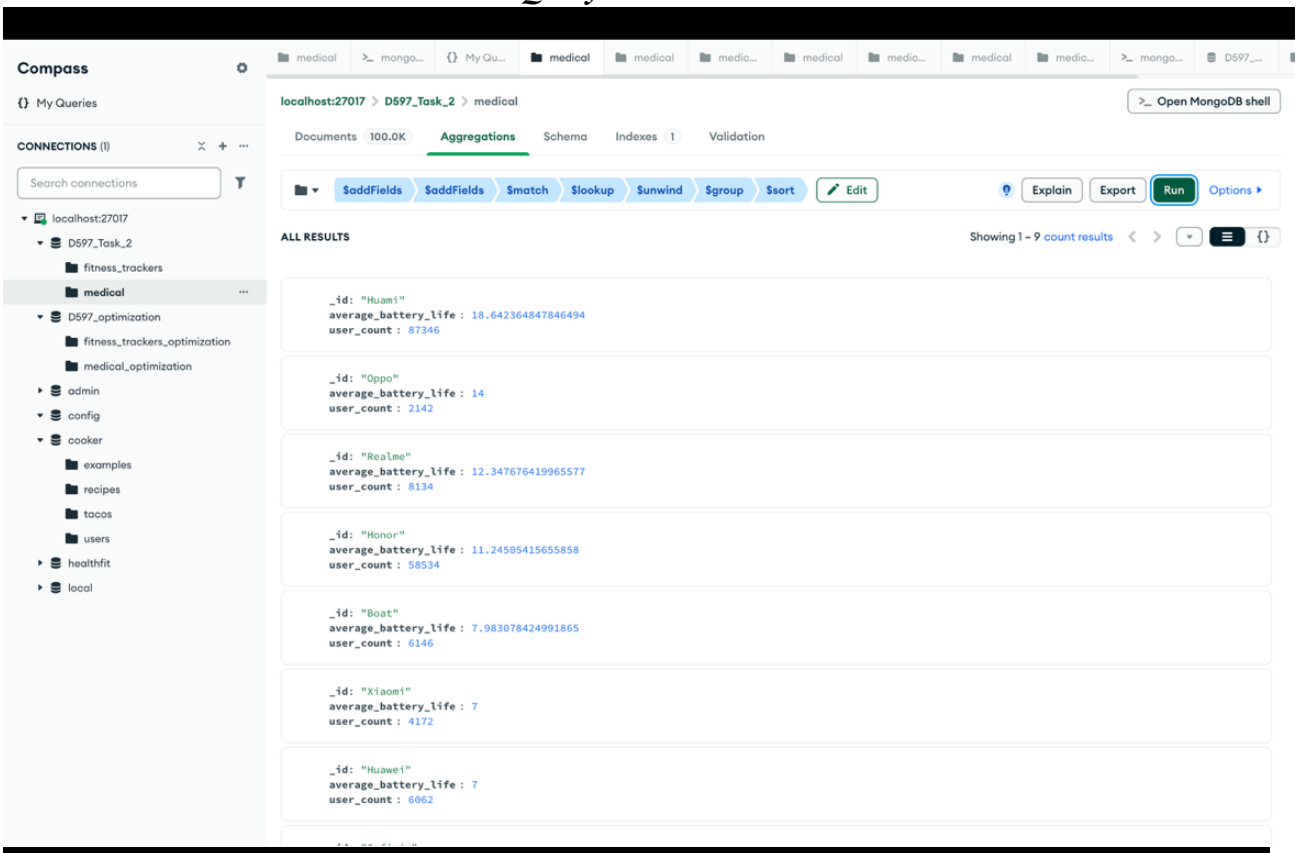
Query3_BatteryLife_by_Tracker_Brand – Stage 5 & 7 (out of 7)

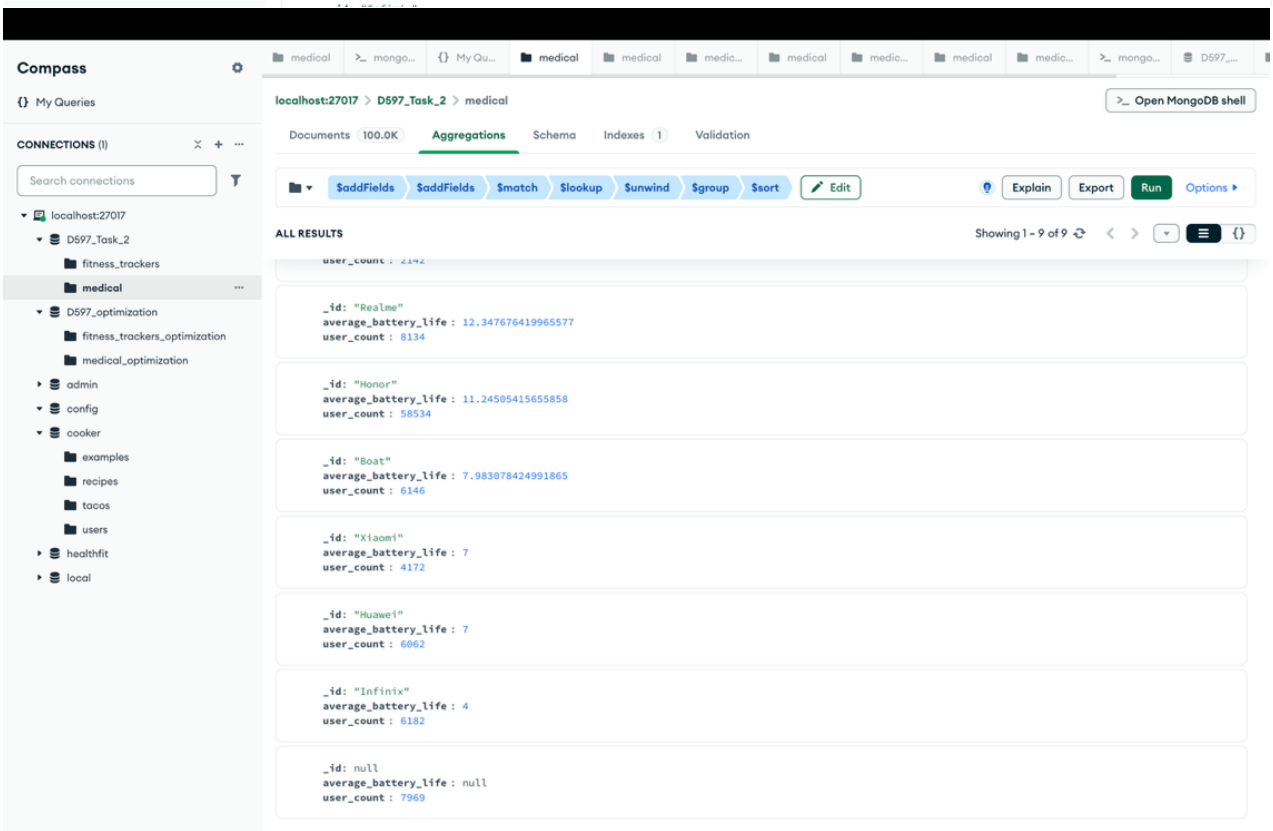


Query3_BatteryLife_by_Tracker_Brand – Stage 7 (out of 7)



Query 3 Results

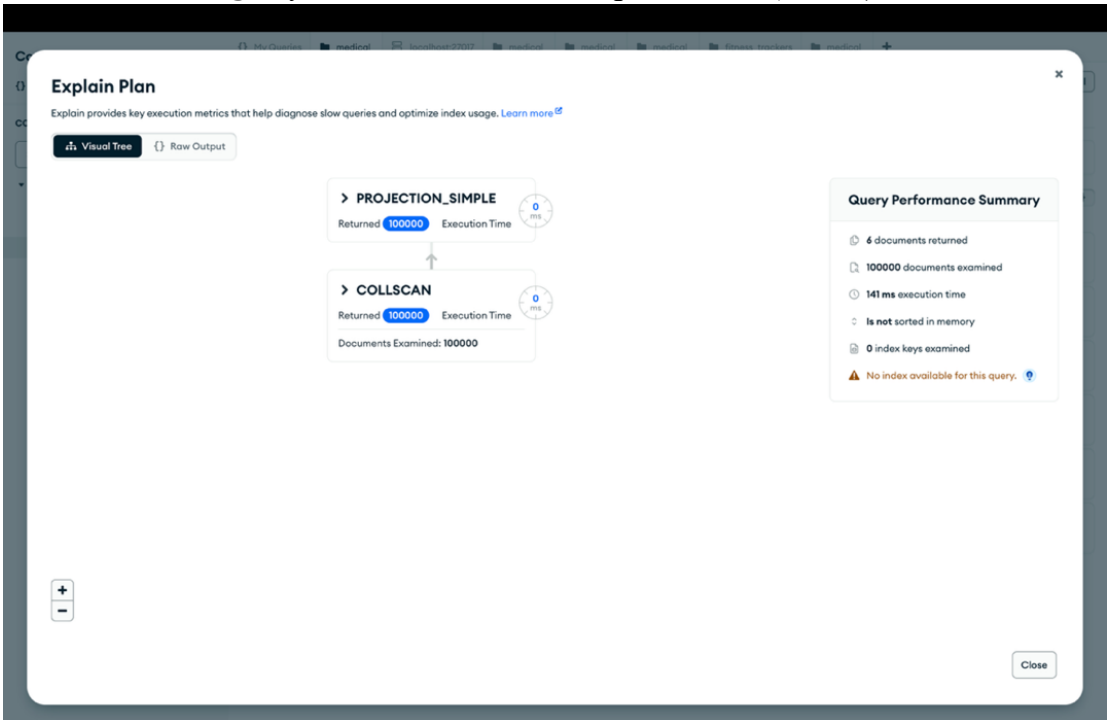




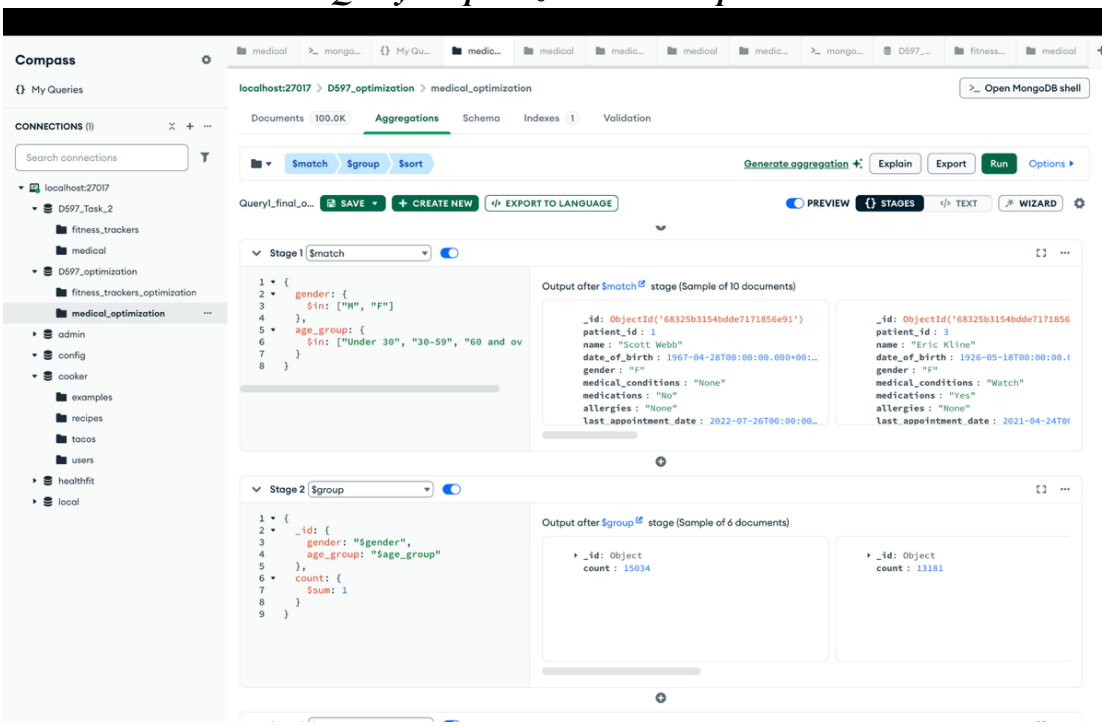
D4. Optimization Techniques

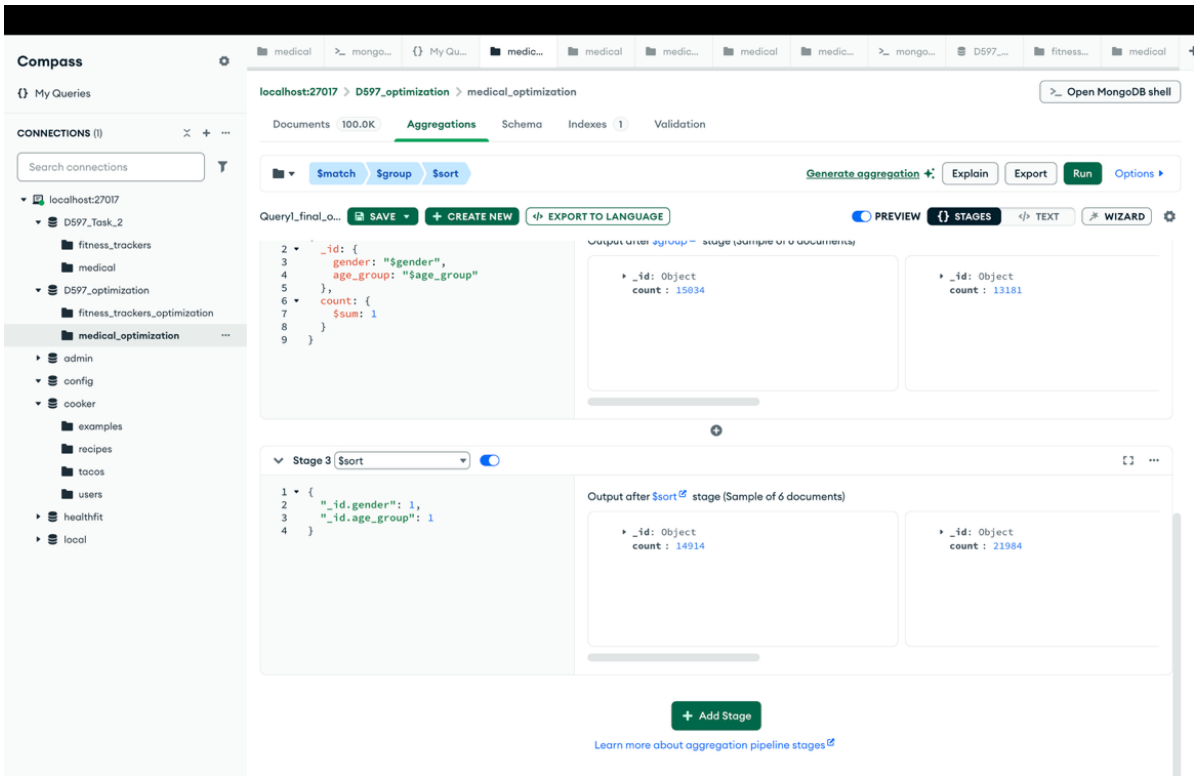
QUERY 1 OPTIMIZATION

Query 1 Execution Without Optimization (141ms)

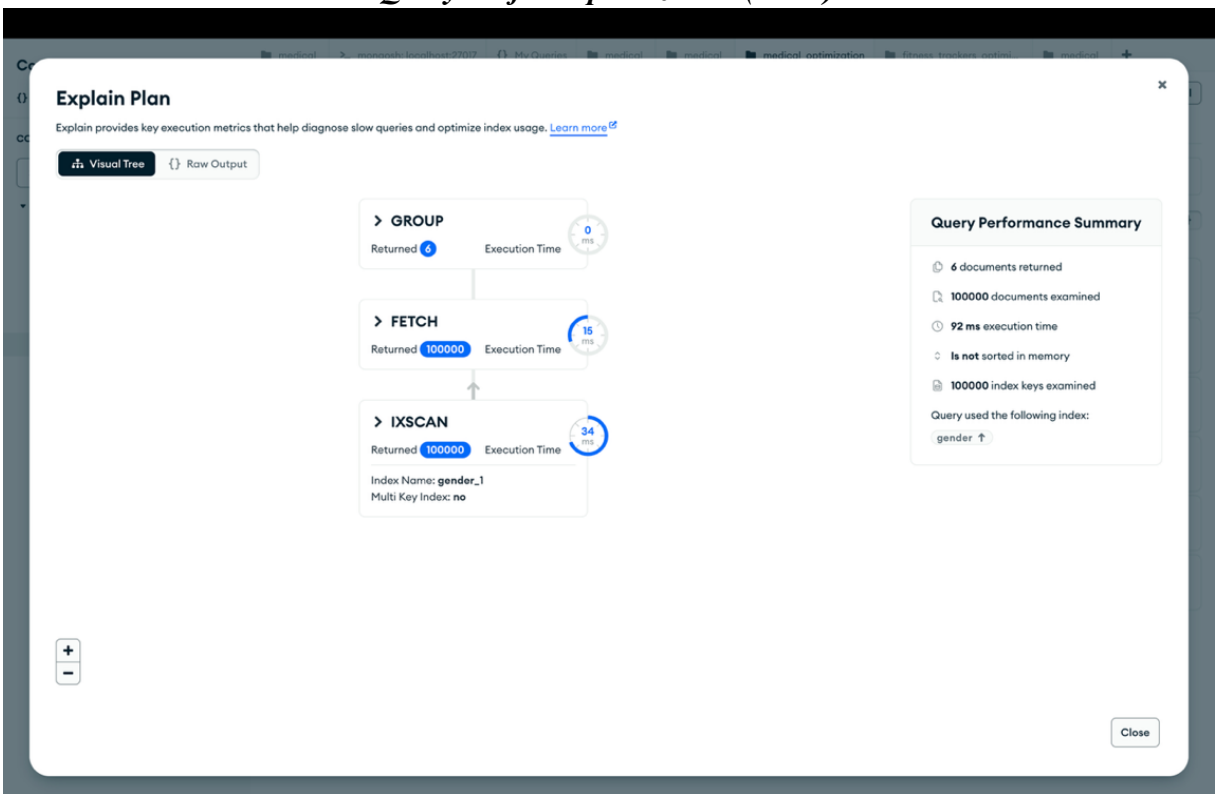


Query 1 Optimization Techniques



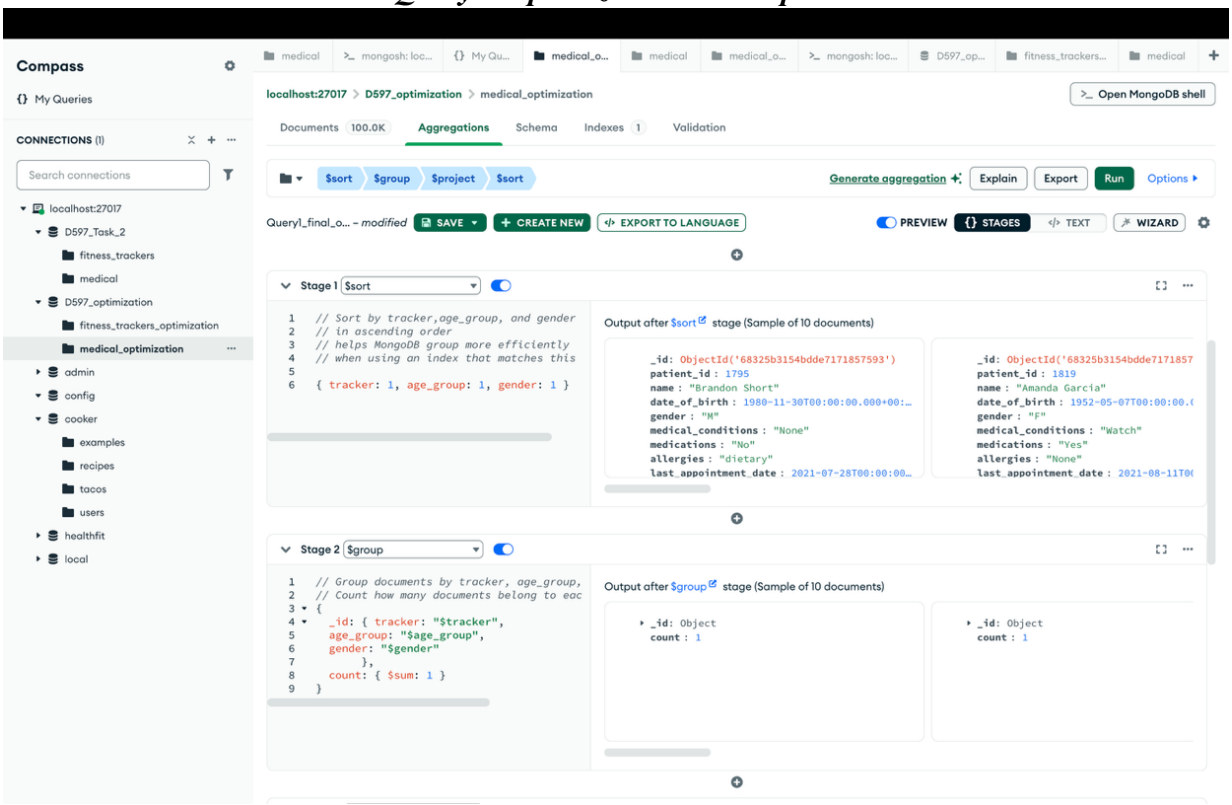


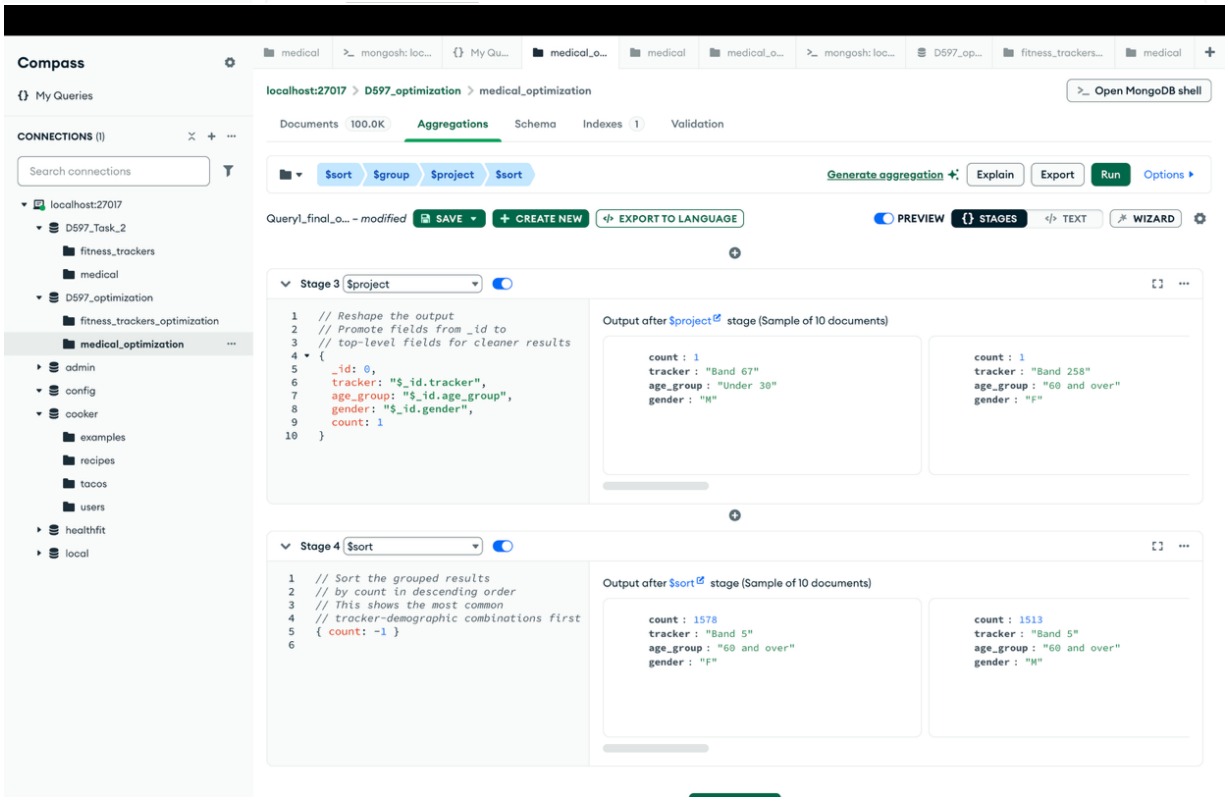
Query 1 After Optimization (92ms)



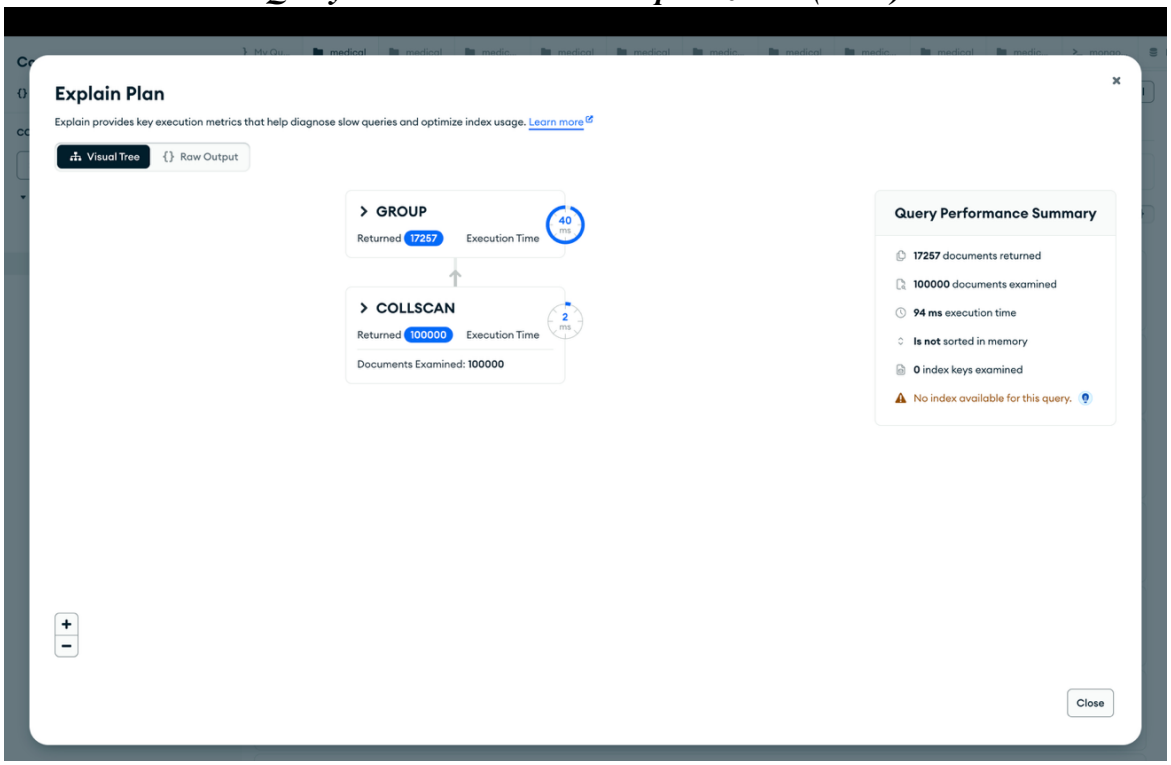
QUERY 2 OPTIMIZATION

Query 2 Optimization Techniques

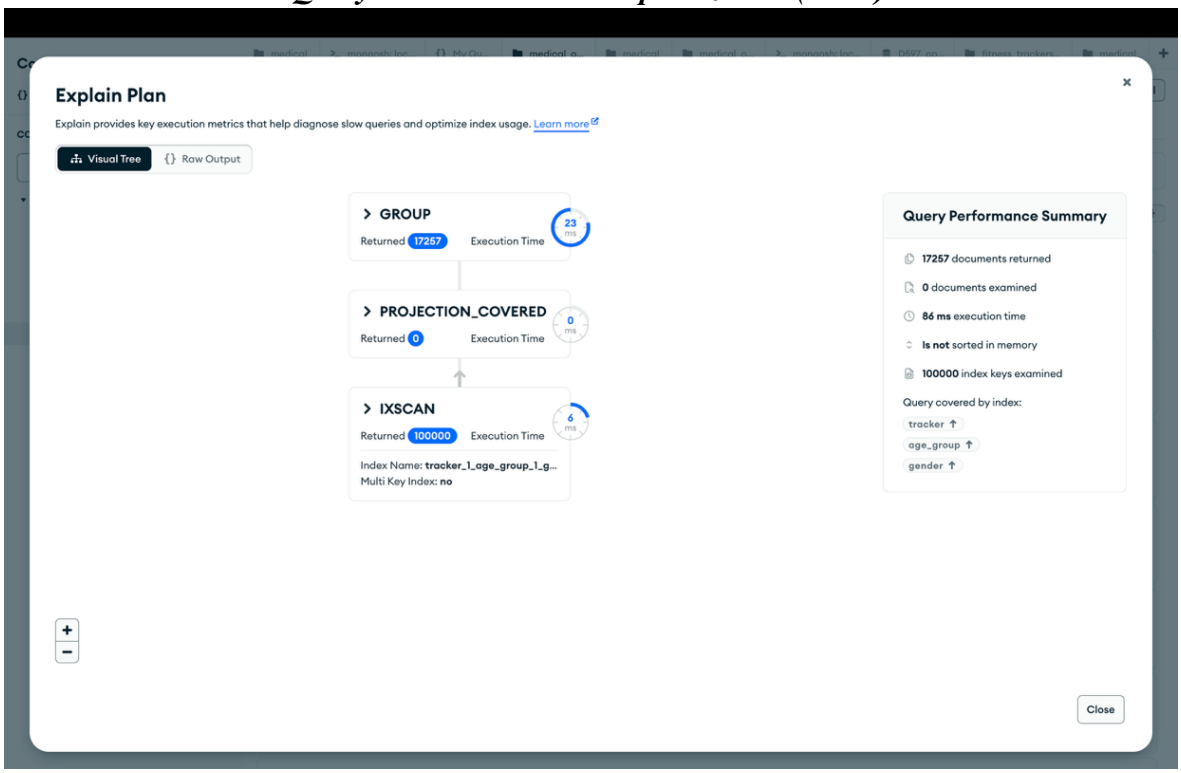




Query 2 Execution Without Optimization (94ms)

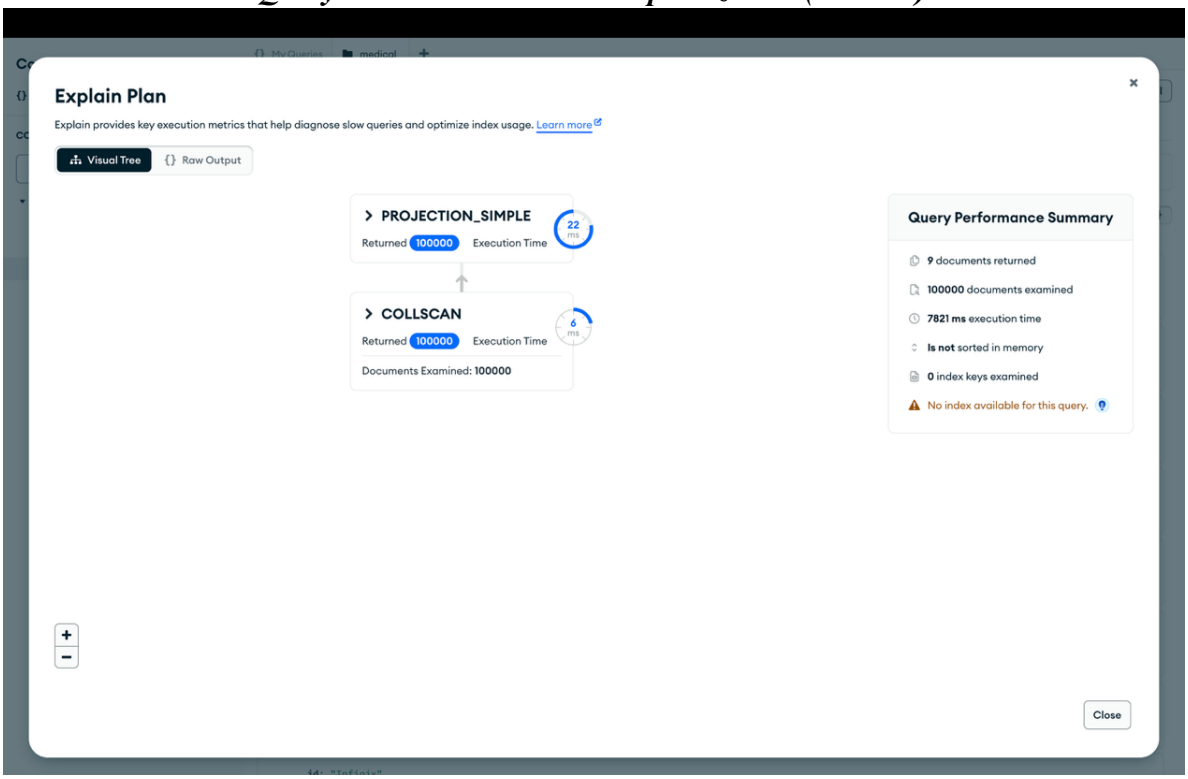


Query 2 Execution With Optimization (86ms)

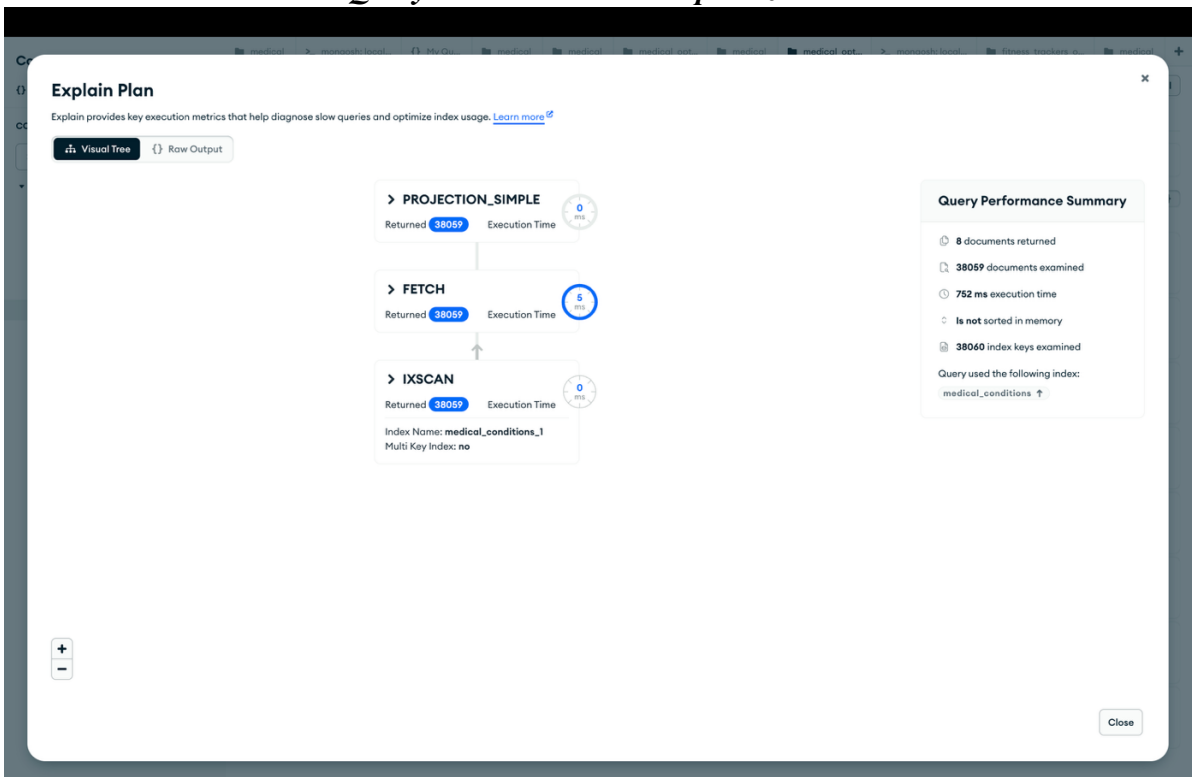


QUERY 3 OPTIMIZATION

Query 3 Execution Without Optimization (7821ms)



Query 3 Execution With Optimization



References

Y. He, S. Lee, R. Zhang, & M. S. Kim. (2021). *Scalable storage of real-time personal health data using NoSQL databases in smart health applications. Sensors*, 21(2), 535. https://doi.org/10.3390/s21020535

M. Dey, M. Kabir, & A. Ghosh (2020). *Evaluating NoSQL databases for scalable real-time health monitoring systems. Journal of Big Data*, 7(1)**.** https://doi.org/10.1186/s40537-020-00320-6

G. Alshammari, A. Alhaidari, & A. Almogren. (2019). *NoSQL for healthcare big data: A performance evaluation of MongoDB and Cassandra. IEEE Access*, **7**, 149848–149855. https://doi.org/10.1109/ACCESS.2019.2946975

Y. Zhang, J. Dang, & L. Zhang. (2019). *Real-time health monitoring and analytics with big data. IEEE Transactions on Industrial Informatics*, 15(1), 384–393. https://doi.org/10.1109/TII.2018.2856580

Fernández-Alemán, J. L., Señor, I. C., Lozoya, P. Á. O., & Toval, A. (2013). *Security and privacy in electronic health records: A systematic literature review*. Journal of Biomedical Informatics, 46(3), 541–562. https://doi.org/10.1016/j.jbi.2012.12.003

MongoDB, Inc. (n.d. 1). *Client-side field level encryption*. MongoDB Documentation. https://www.mongodb.com/docs/manual/core/security-client-side-encryption/

Gajanayake, R., Iannella, R., & Sahama, T. (2017). Security and privacy in the use of cloud computing for health data. *Health Information Science and Systems, 5*(1). https://doi.org/10.1007/s13755-017-0020-5

MongoDB, Inc. (n.d. 2). *Replication*. MongoDB Documentation. https://www.mongodb.com/docs/manual/replication/

Xie, H., Wang, F., & Zhang, L. (2020). Data masking approaches for privacy-preserving electronic health record sharing. *BMC Medical Informatics and Decision Making, 20*(Suppl 3), 132. https://doi.org/10.1186/s12911-020-01125-4

MongoDB, Inc. (n.d. 3). *Security checklist*. MongoDB Documentation. https://www.mongodb.com/docs/manual/administration/security-checklist/

Dey, M., Kabir, M. A., & Ghosh, A. (2020). Evaluating NoSQL databases for scalable real-time health monitoring systems. *Journal of Big Data, 7*(1), 1–25. https://doi.org/10.1186/s40537-020-00320-6

Wolniak, R. (2023). Functioning of real-time analytics in business. *Scientific Papers of Silesian University of Technology: Organization and Management Series*, (172), 659–672. https://doi.org/10.29119/1641-3466.2023.172.40

Belefqih, S., Zellou, A., & Berquedich, M. (2024). Semantic schema extraction in NoSQL databases using BERT embeddings. *Data Science Journal*, *23*(1), 1–15. https://doi.org/10.5334/dsj-2024-057