Cloud poe part 3

ST10053561

Varsity College Cape town

BCAD

**Table of Contents**

[List the Azure components utilised thus far in the POE 3](#_Toc151110282)

[Motivation for Needed Change 3](#_Toc151110283)

[Necessary Changes for Part 2 And Testing the Code 4](#_Toc151110284)

[References 9](#_Toc151110285)

[Figure 1 : Updated Function code 1 by Feedback 3](#_Toc151127903)

[Figure 2 : Function Code 2 : Retrieving Values from Queue to Azure Cosmos DB 3](#_Toc151127904)

[Figure 3 : Azure Cosmos DB Main Code to Send Data 4](#_Toc151127905)

[Figure 4 : Console Application Message 1 4](#_Toc151127906)

[Figure 5 : Console Application Message 2 5](#_Toc151127907)

[Figure 6 : Console Application Message 3 5](#_Toc151127908)

[Figure 7 : Console Application Message 4 5](#_Toc151127909)

[Figure 8 : Console Application Message 5 5](#_Toc151127910)

[Figure 9 : Five Messages in Queue in Azure Portal 5](#_Toc151127911)

[Figure 10 : Queue Messages in Azure Cosmos DB Part 1 6](#_Toc151127912)

[Figure 11 : Queue Messages in Azure Cosmos DB Part 2 6](#_Toc151127913)

[Figure 12 : Queue Messages in Azure Cosmos DB Part 3 7](#_Toc151127914)

# List the Azure components utilised thus far in the POE

|  |  |  |
| --- | --- | --- |
| **Component** | **Technology Choice** | **Hosting Model** |
| Azure SQL database | Data storage | PAAS |
| Azure Storage Queue | Data storage | PAAS |
| Visual Studio | Compute | SAAS |

# Motivation for Needed Change

To meet the primary requirements for ‘Aweh Productions’ I would like to change component of ‘Azure SQL database’ to ‘Azure Cosmos DB’ to match the primary requirements. Here are the reasons for why I want to choose the ‘Azure Cosmos DB’ based on (Seesharprun, 2023).

1. Fast Data Retrieval: This Requirement ensures that users can access their vaccination information quickly and efficiently, it is crucial that the retrieval process be optimized for speed. Azure Cosmos DB stands out in this regard, delivering single-digit millisecond response times at the 99th percentile globally. This exceptional performance ensures that users worldwide can effortlessly access their data, regardless of their location. This capability directly addresses the need for rapid retrieval of vaccination information, making it an ideal choice for this critical application.

2. Diverse Data Formats: This requirement has ability to store different data formats provided by different data manufacturers. Azure Cosmos DB is a multi-model database service, which means it natively supports multiple data models, including document, key-value, graph, and column-family. Its schema-agnostic indexing engine can automatically index all the data it ingests without requiring any schema or secondary indexes from the developer. This flexibility makes it an excellent choice for storing diverse data formats.

3. Cost Efficiency: The final requirement is to keep the operational costs of running the solution as low as possible. Azure Cosmos DB can help achieve this through its provisioned throughput model. You can specify the throughput you need for your workload, and Azure Cosmos DB will handle the rest, scaling up and down as needed. This way, you only pay for the throughput you need. Additionally, with Azure Cosmos DB's global distribution, you can scale out your database to any number of Azure regions worldwide with a single click, allowing you to achieve massive scale and global distribution without worrying about budget-breaking costs.

In conclusion, replacing my Azure SQL Database with Azure Cosmos DB could provide a more efficient and cost-effective solution that meets all of ‘Aweh Productions' primary requirements. It's a change that could potentially bring significant benefits to the company's operations.

# Necessary Changes for Part 2 And Testing the Code

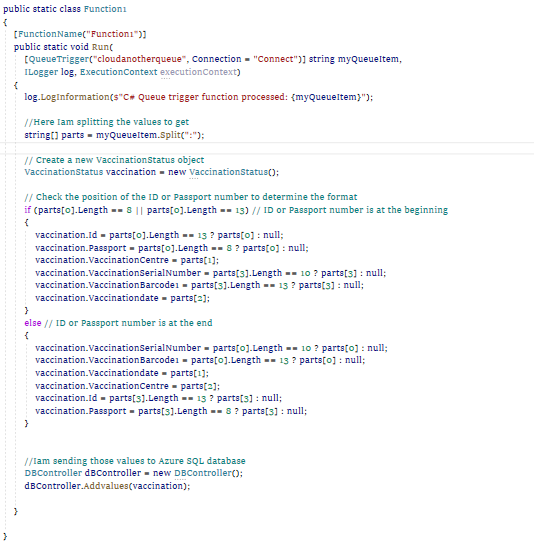


Figure 1 : Updated Function code 1 by Feedback

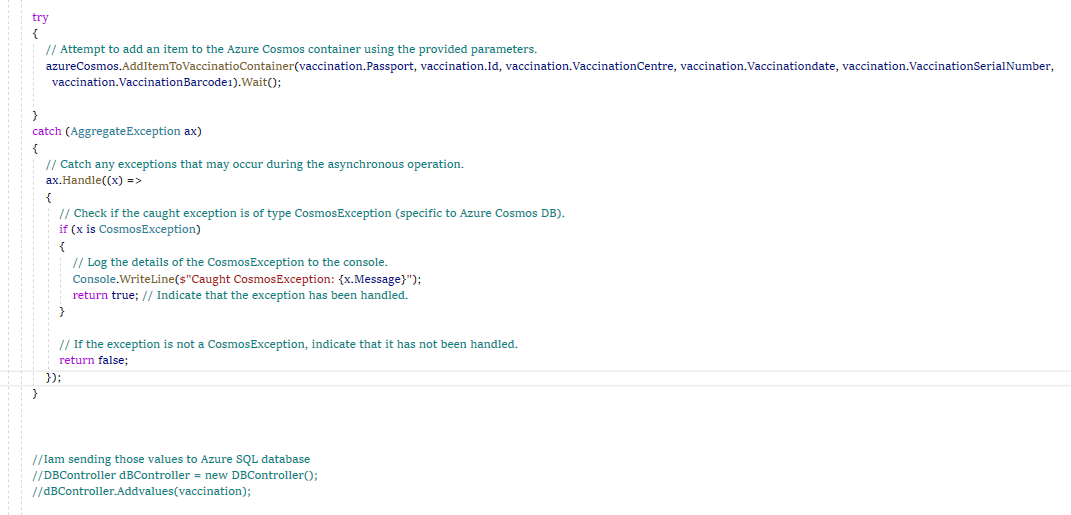


Figure 2 : Function Code 2 : Retrieving Values from Queue to Azure Cosmos DB



Figure 3 : Azure Cosmos DB Main Code to Send Data

# Sending Messages To Queue

A screen shot of a computer

Description automatically generated

Figure 4 : Console Application Message 1

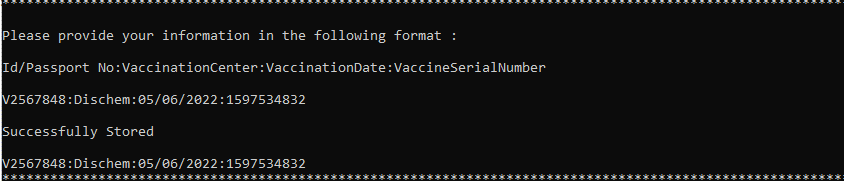


Figure 5 : Console Application Message 2

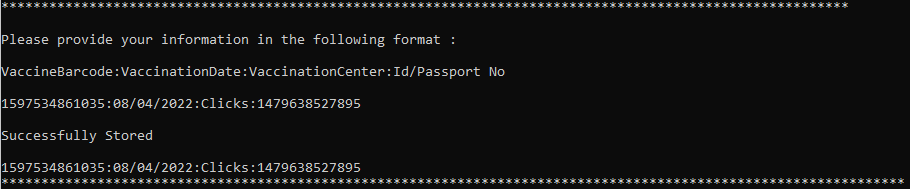


Figure 6 : Console Application Message 3

A screen shot of a computer

Description automatically generated

Figure 7 : Console Application Message 4

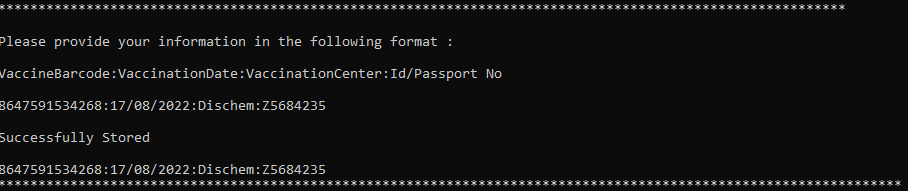


Figure 8 : Console Application Message 5

# Storing Messages in Queue

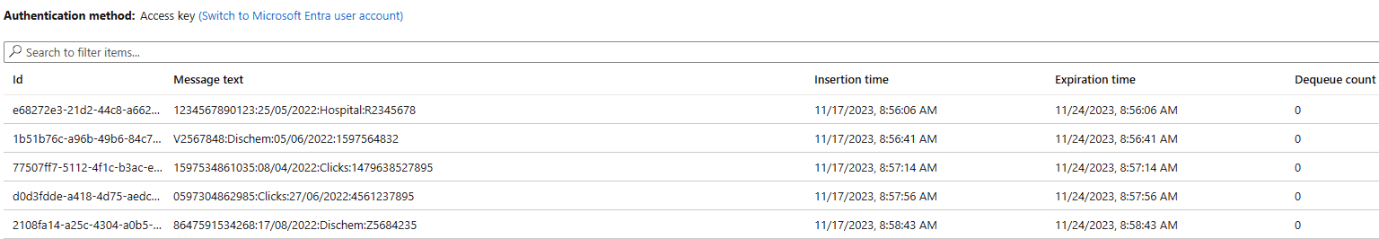


Figure 9 : Five Messages in Queue in Azure Portal

# Storing Messages in Azure Cosmos DB

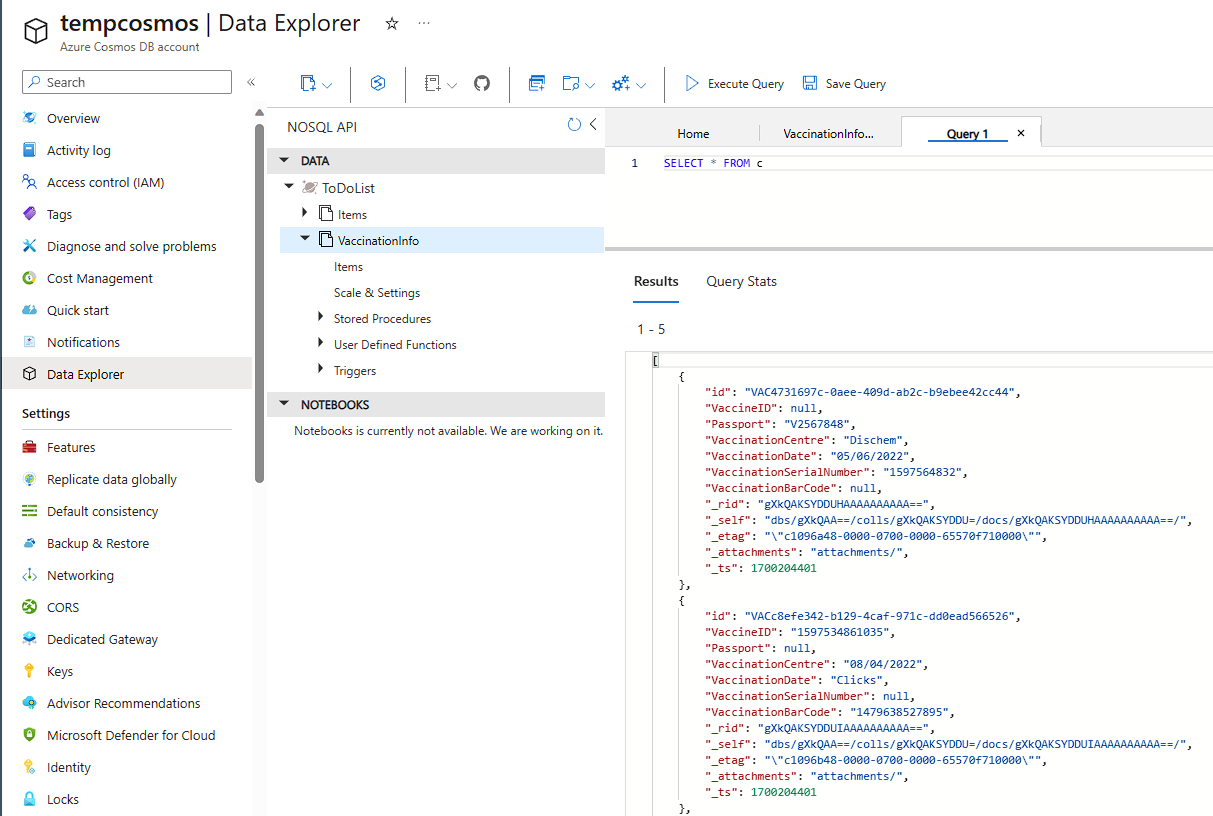


Figure 10 : Queue Messages in Azure Cosmos DB Part 1

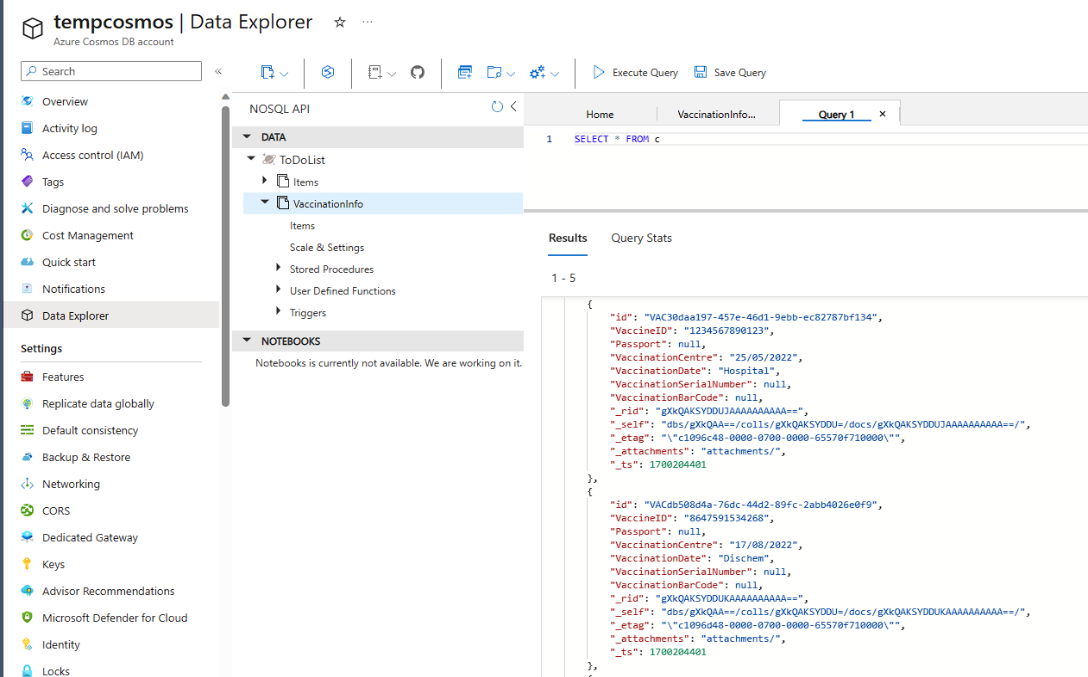


Figure 11 : Queue Messages in Azure Cosmos DB Part 2

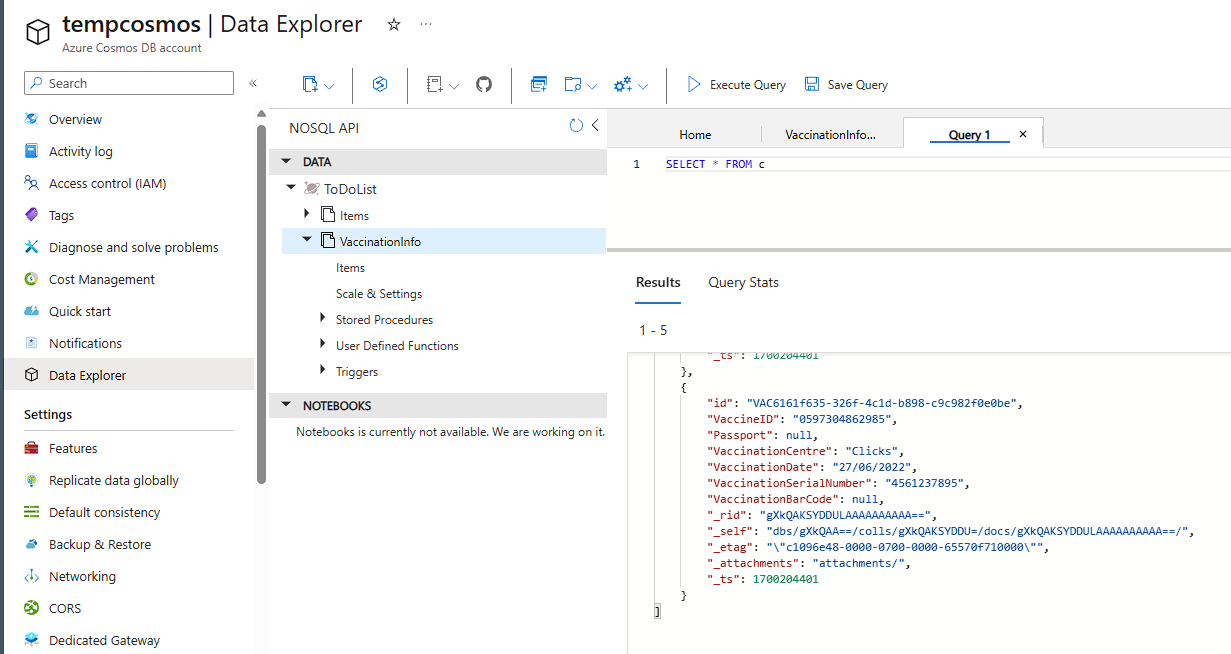


Figure 12 : Queue Messages in Azure Cosmos DB Part 3

# References

*Create Cosmos DB Container & Add Items To Container.* 2020. [Film] s.l.: Keshav-Learn IT-Self.

Seesharprun, 2022. *Get started with Azure Cosmos DB for nosql using .NET.* [Online]   
Available at: https://learn.microsoft.com/en-us/azure/cosmos-db/nosql/how-to-dotnet-get-started?tabs=azure-portal%2Cwindows  
[Accessed 17 November 2023].

Seesharprun, 2023. *Azure cosmos DB – unified AI database.* [Online]   
Available at: https://www.citethisforme.com/cite/sources/websiteautociteconfirm  
[Accessed 15 November 2023].