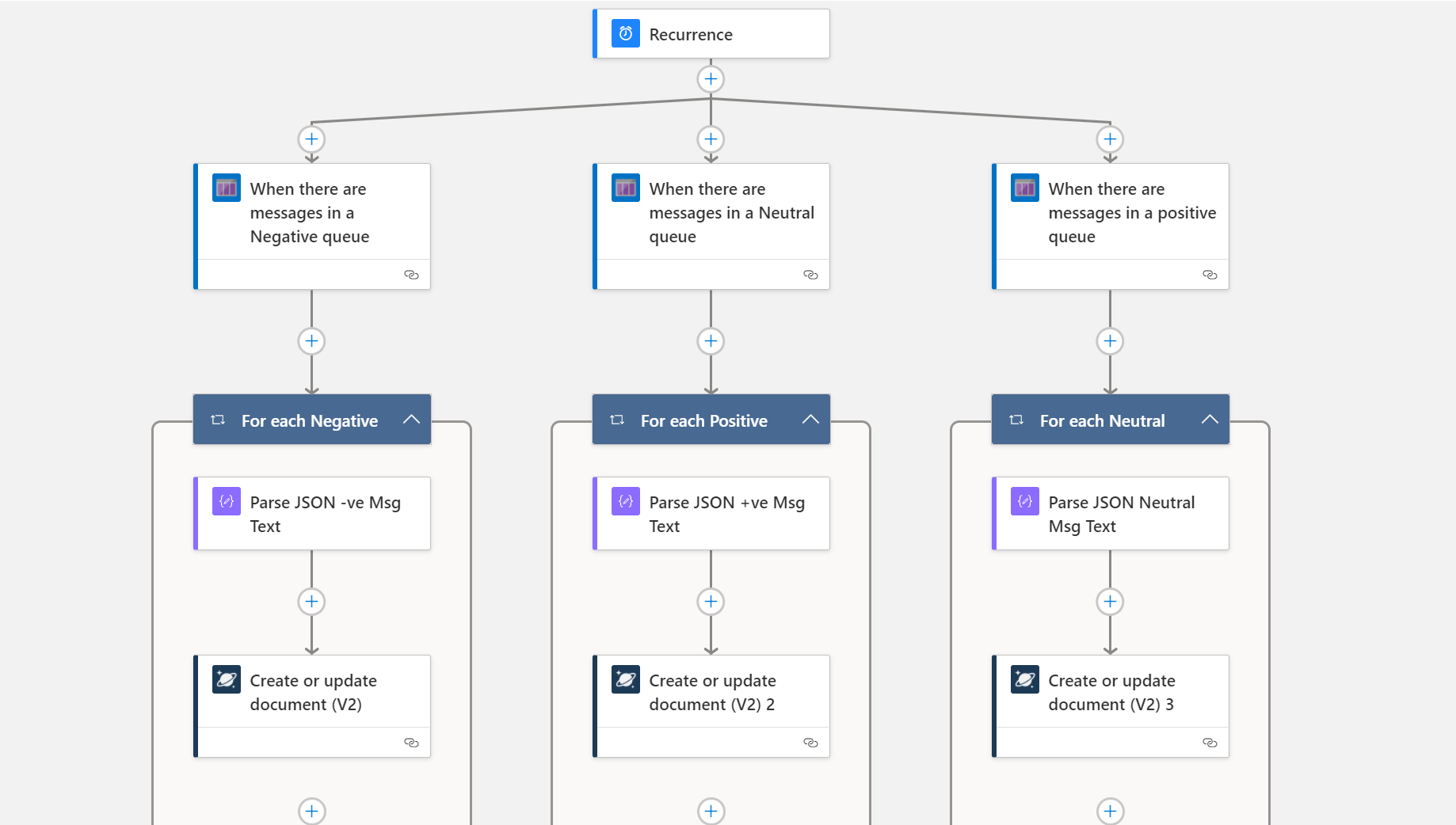
**Sentiment Analysis with Storage Queues, Azure Function, Logic Apps, Cosmos DB, and Power BI**

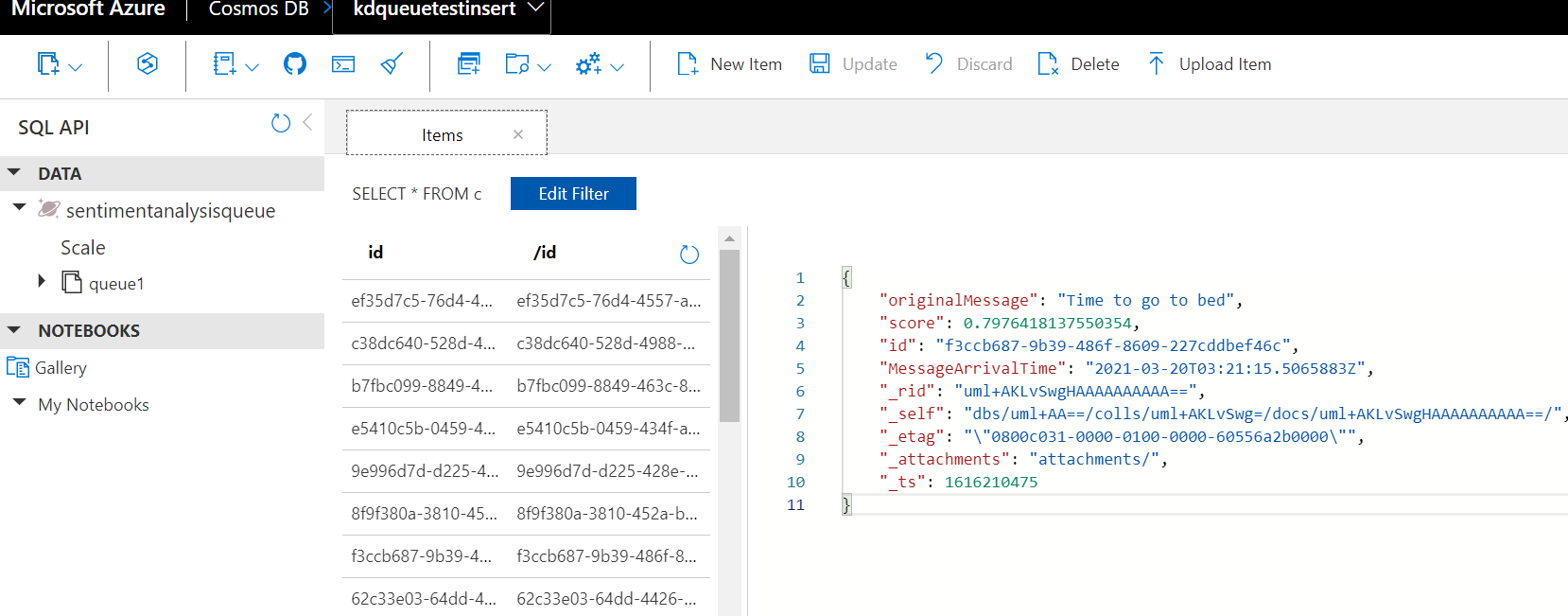
**High-Level Overview of the solution**



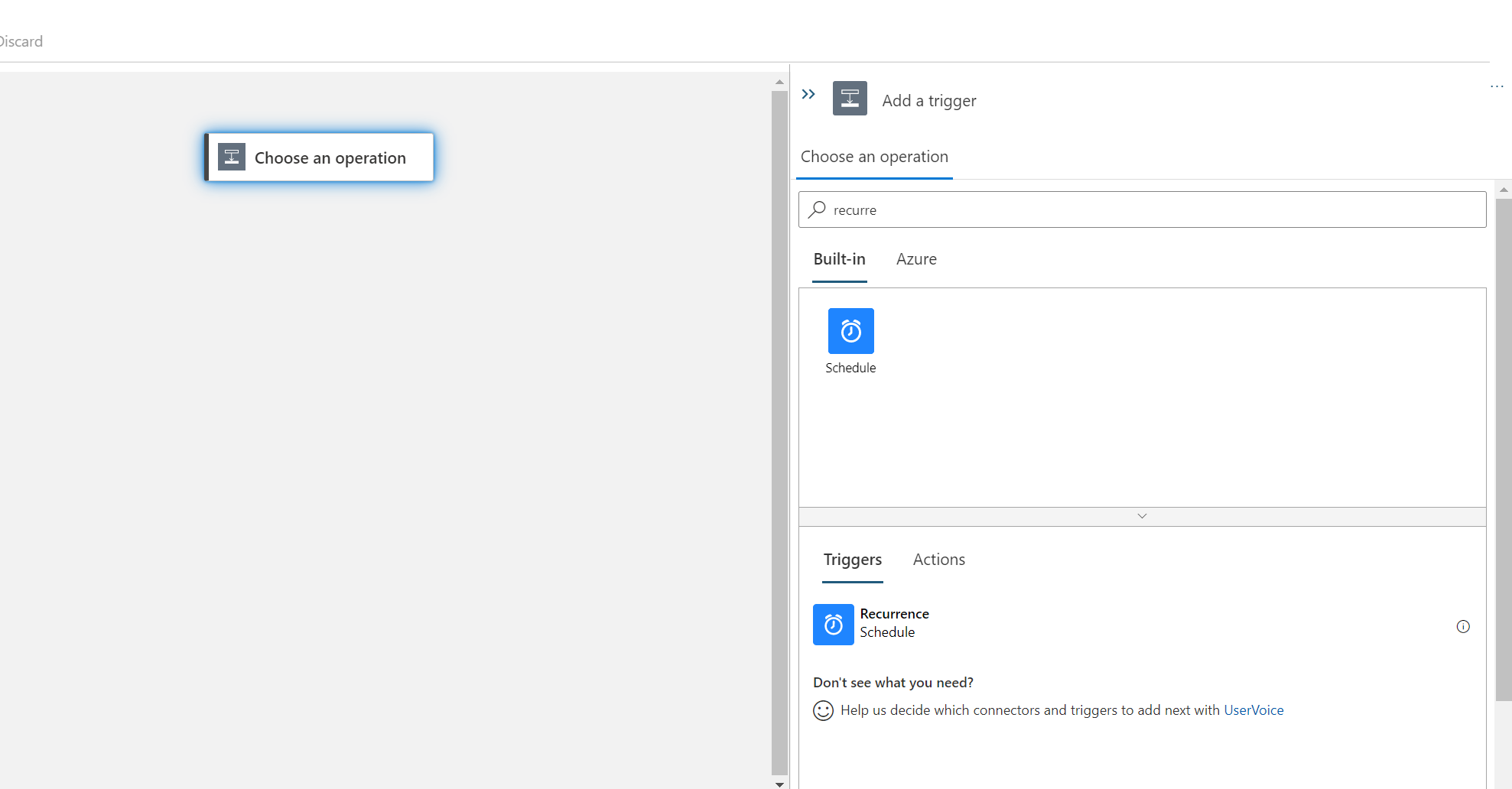
1. In the [Azure portal](https://portal.azure.com/), create an empty logic app by following the instructions in [Create your logic app](https://docs.microsoft.com/en-us/azure/logic-apps/quickstart-create-first-logic-app-workflow#create-your-logic-app). When you see the **Logic Apps Designer**, return to this tutorial.
2. Once you are done with the logic app, it will look like this

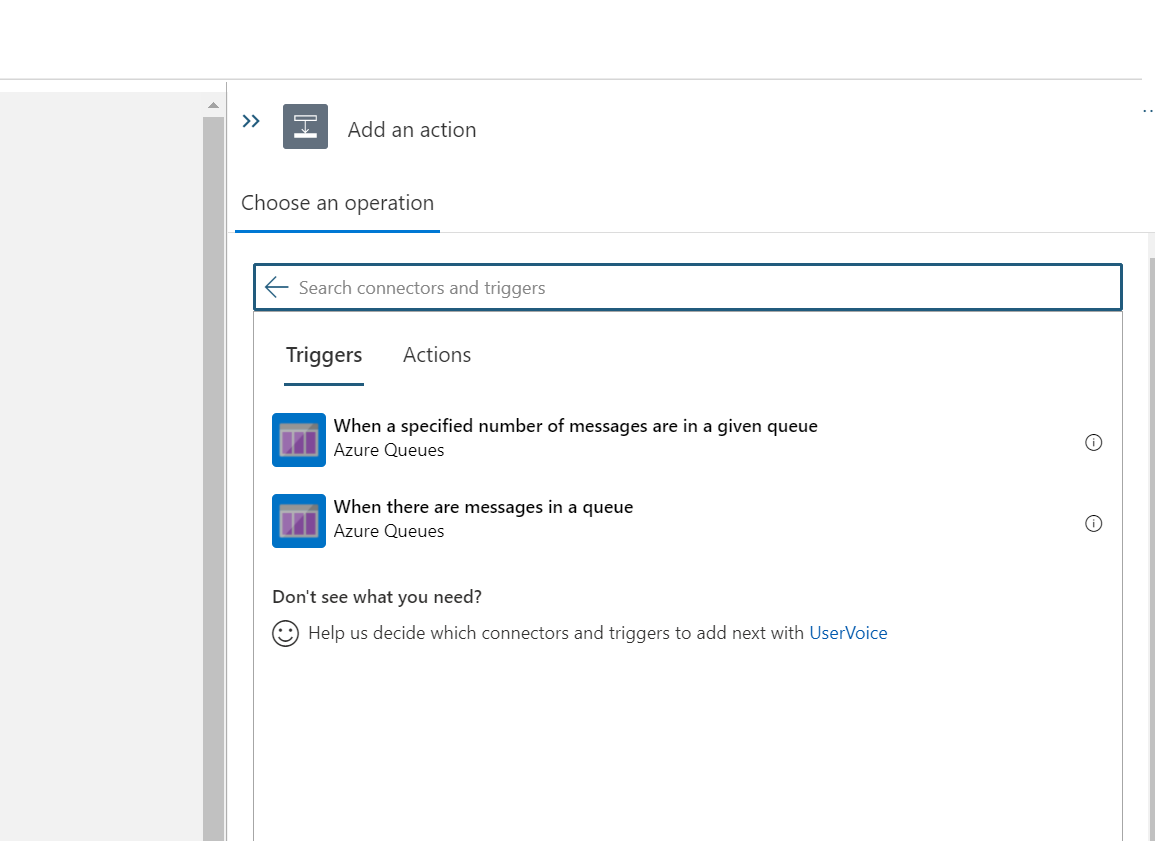


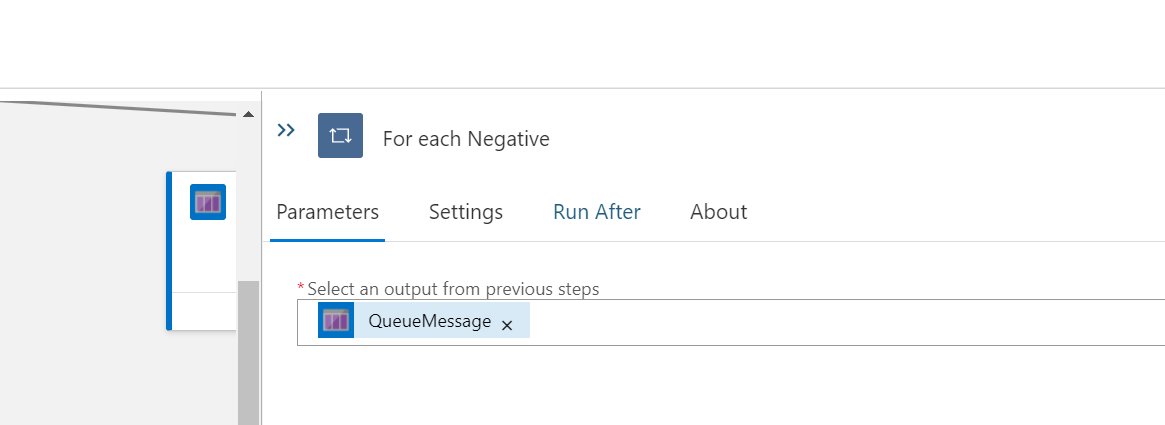
1. The cosmos DB account and database have already been created. You have to create the collection with the name of your choice. For example, I created a collection with the name queue1.



1. Make sure you select stateful workflow in the logic app, not the stateless. In the Logic Apps Designer, add recurrence action. It will run will logic app based on the time interval we want it to run.



1. Select trigger “when there are messages in the queue” and provide the queue name; for example, For positive messages, provide queue name positive-feedback-q. Add a parallel step by clicking + sign and create the trigger for negative and neutral queue also.
2. Add “For Each” loop as we have to process all arrived messages in the queue. Inside ‘For Each’ add ParseJson action. As queue messages are stored as JSON, we have to parse and select the fields we want for our analysis. We are ignoring some default fields added by queue metadata.
3. Add “queuemessage” from the previous step as a parameter in For Each loop



1. For ParseJSON, select the message text and Provide the below schema

{

    "properties": {

        "originalMessage": {

            "type": "string"

        },

        "score": {

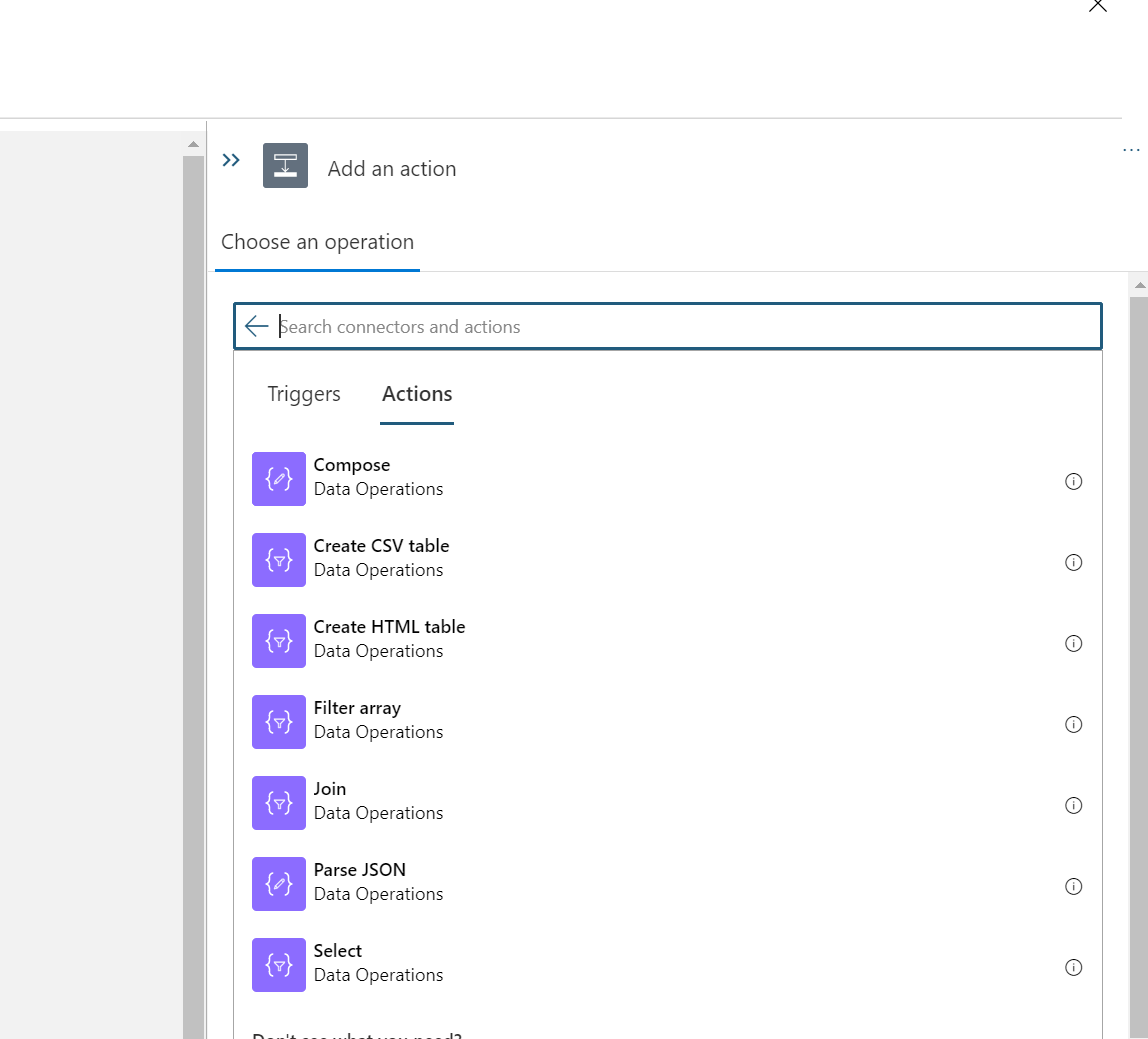
            "type": "number"

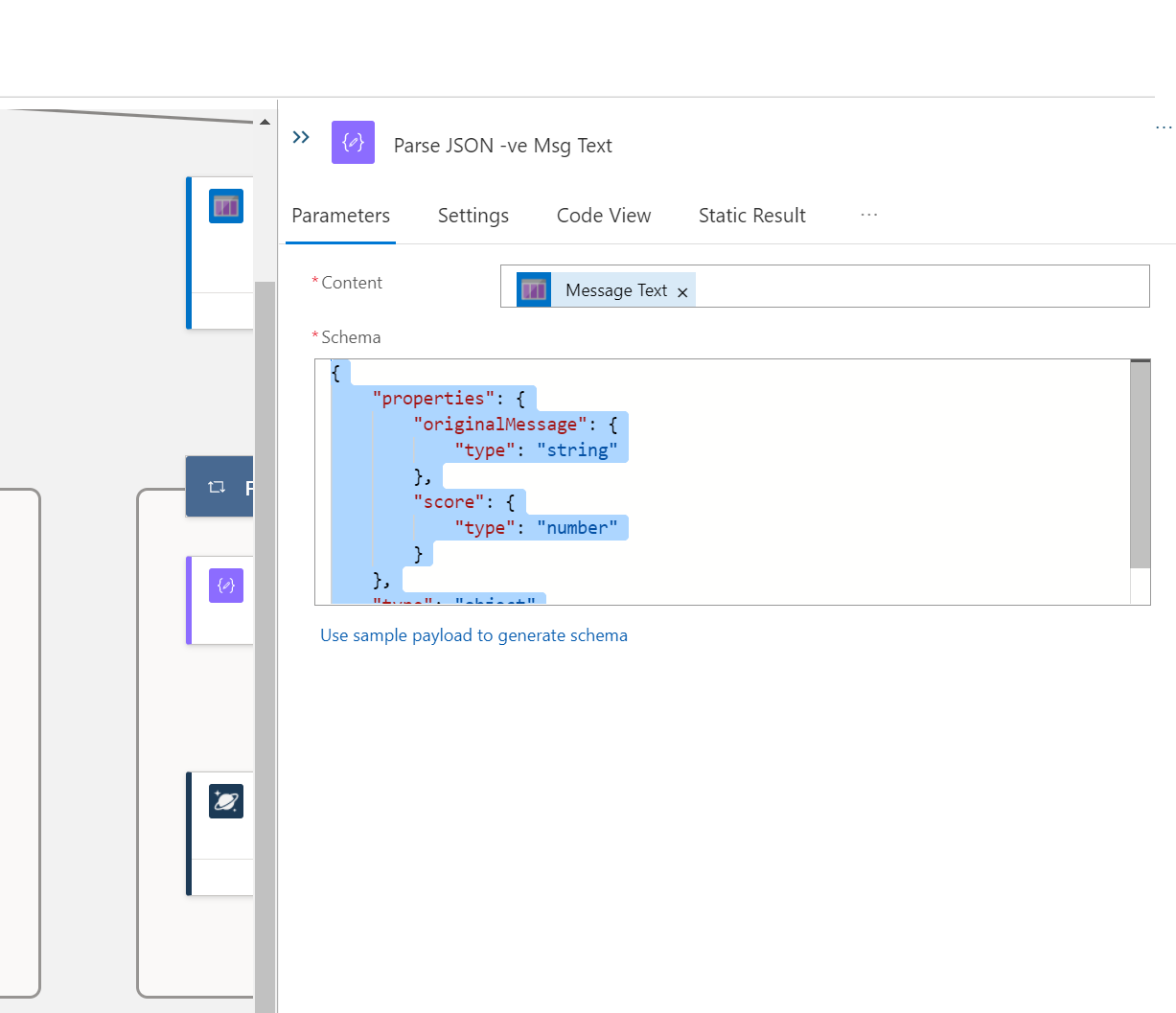
        }

    },

    "type": "object"

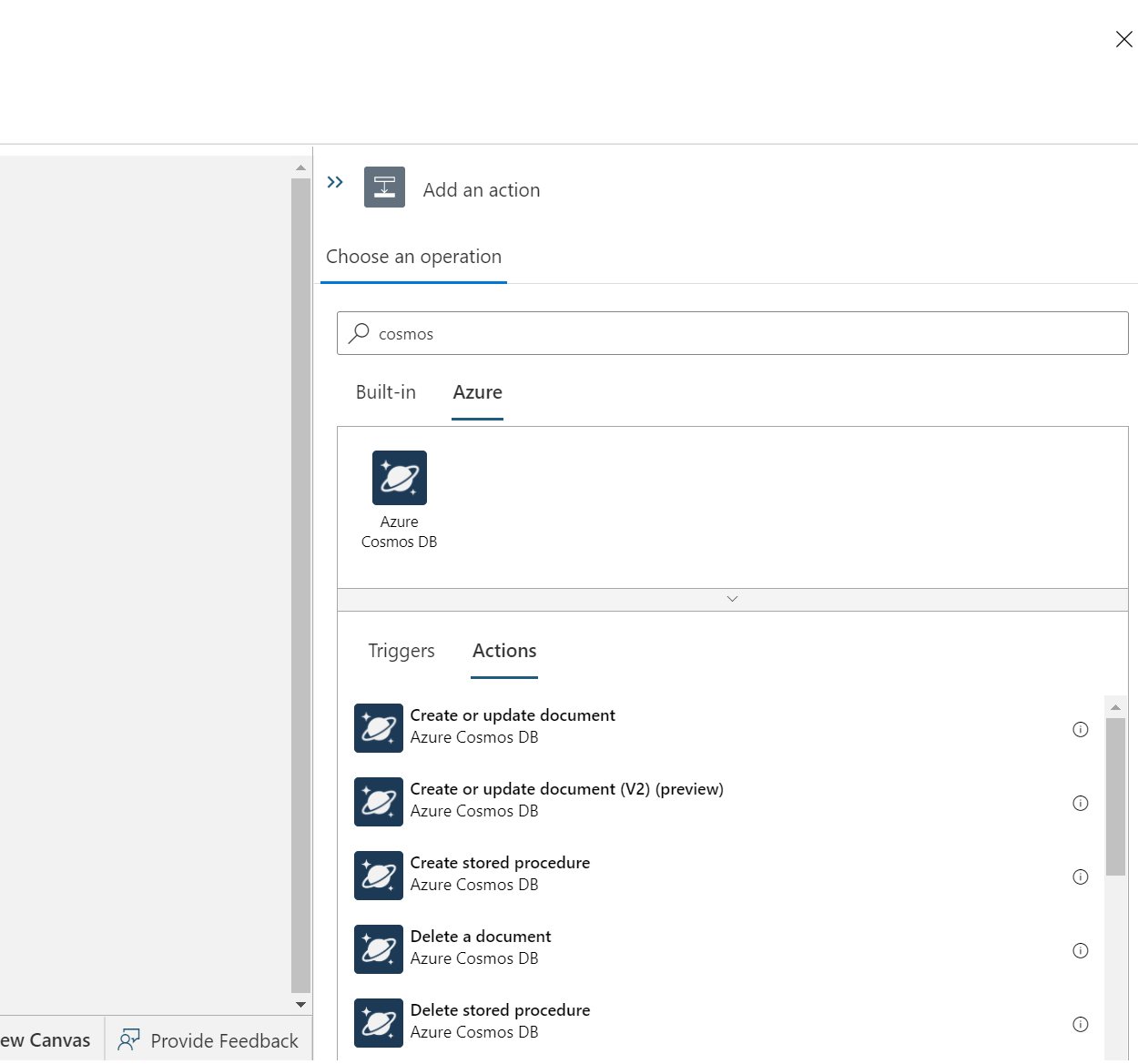
}





Complete these steps for all three message types.

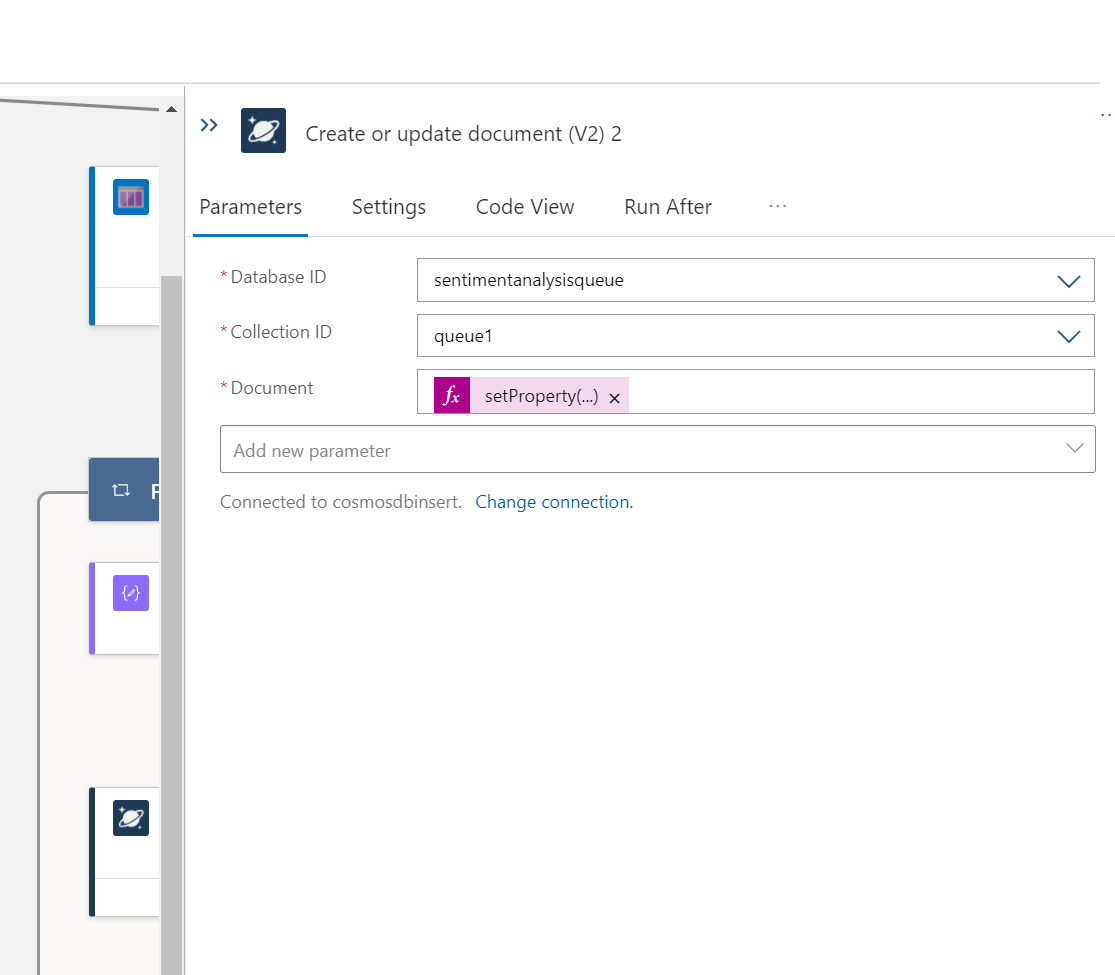
1. Inside ‘For Each’ after ParseJSON action, add ‘Create or update document (V2) (preview)’ to insert messages into cosmos DB collection.



1. In parameter for ‘Create or update document (V2) (preview)’ Select database ID as “sentimemntanalysisqueue” and Collection as queue1 and under document user dynamic content and function to insert message and add ID field for uniqueness. The formula looks like this:  
     
   LOGICAPPS (PREVIEW) Version:

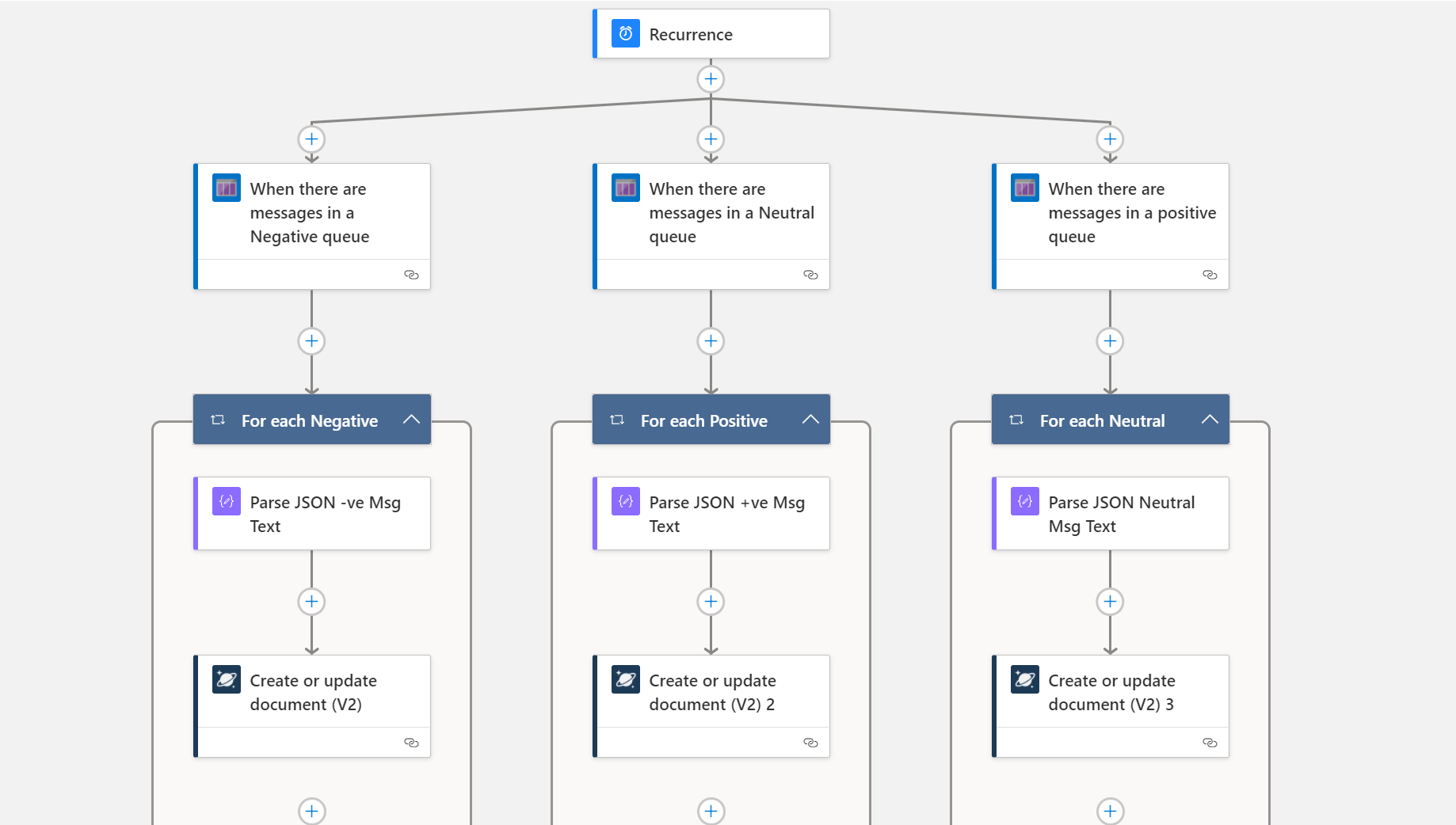
setProperty(setProperty(body('Parse\_JSON\_+ve\_Msg\_Text'),'id',guid()),'MessageArrivalTime',utcNow())

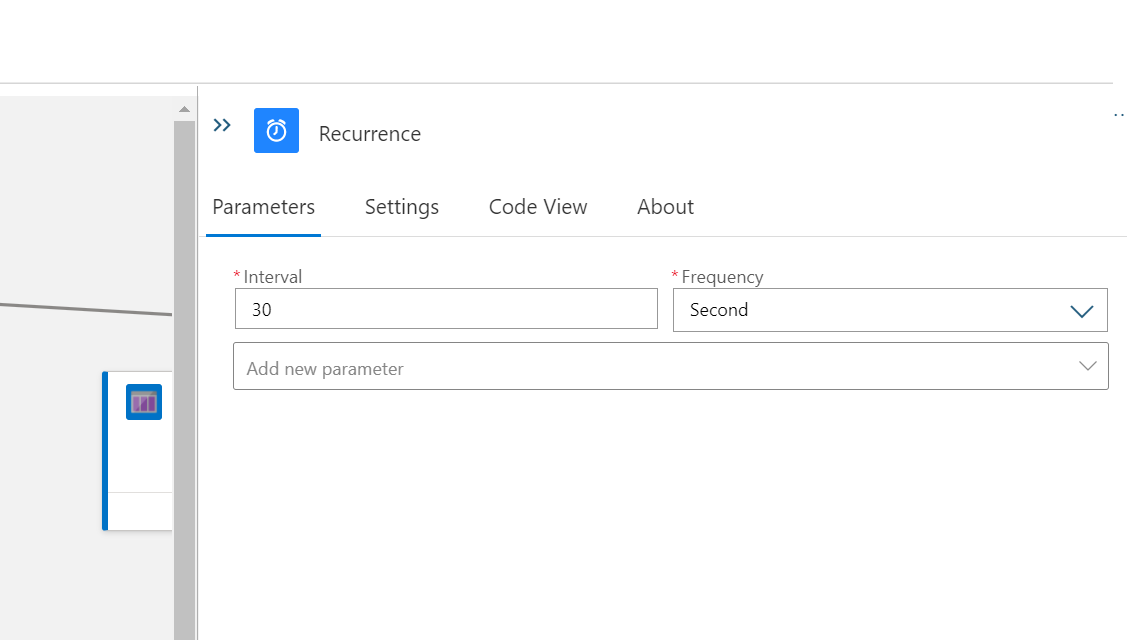
LOGICAPPS (GA) Version: (as of 3/22/2021)  
  
setProperty(setProperty(body('Parse\_JSON\_+ve\_Msg\_Text'),'id',guid()))

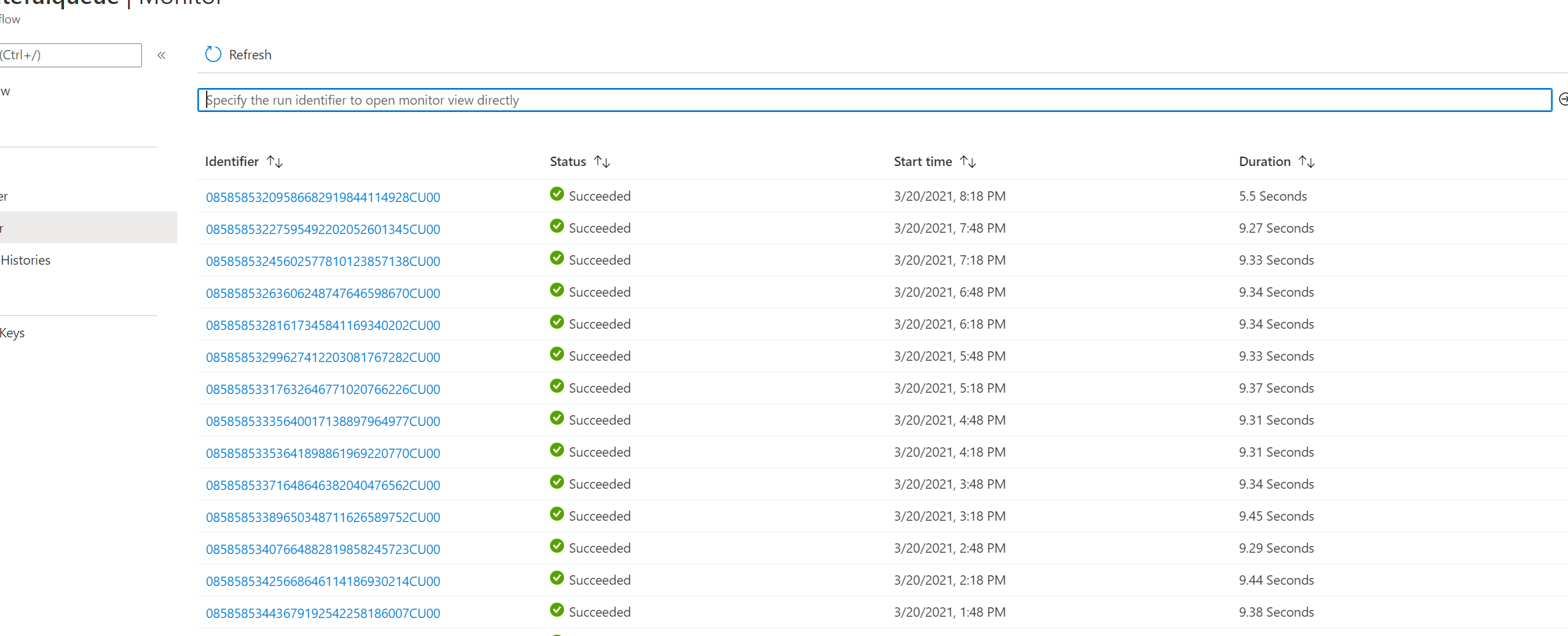


Complete above cosmos DB steps for all three message types.

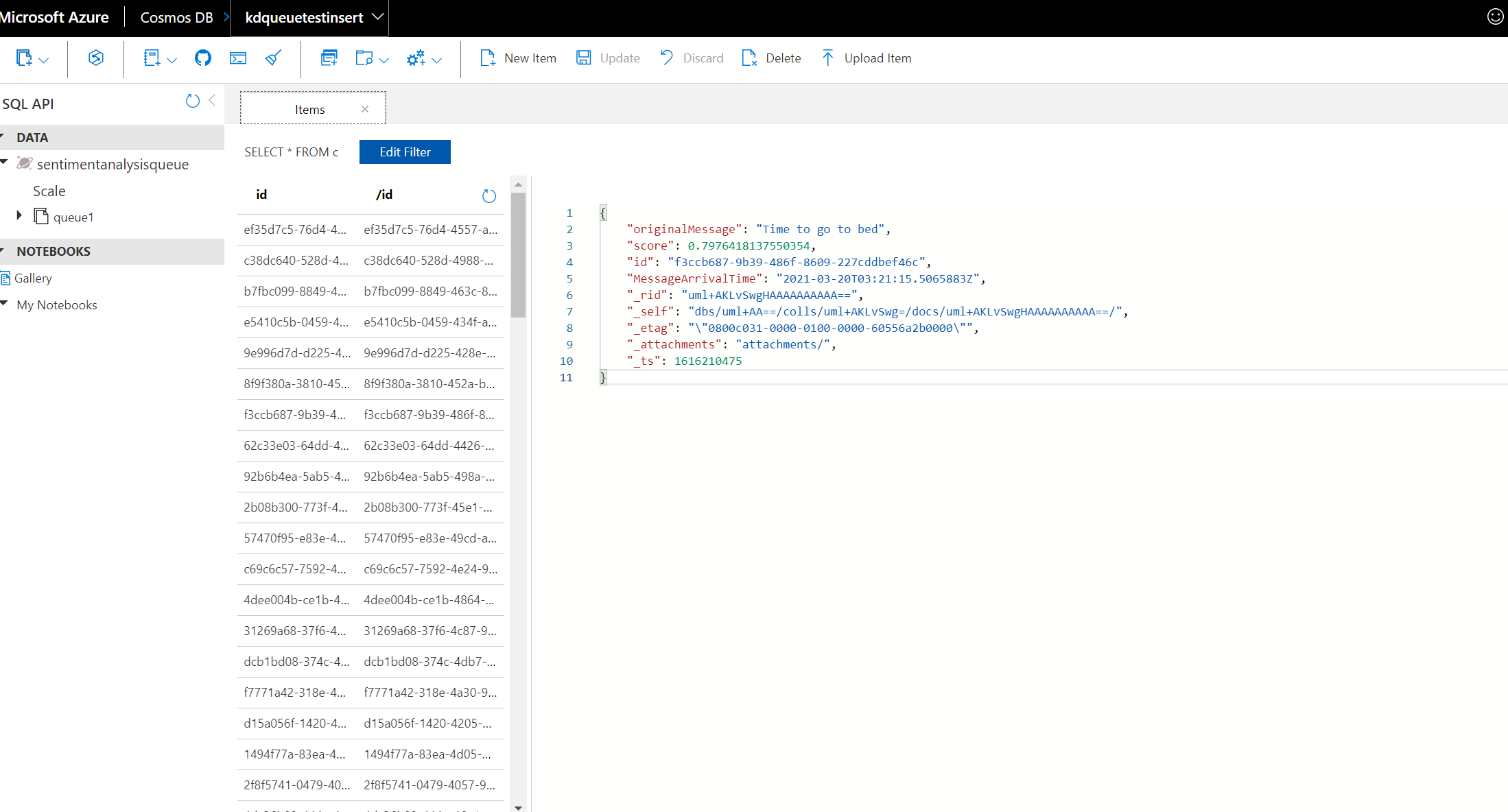
1. Once you are done with step 8, your logic app should look like this



1. If you want to change the recurrence time, then go to recurrence trigger. 
2. Save the Logic App and Monitor its run by going to monitor on the left side.



1. You can to CosmosDB and look for messages inserted in the collection.



1. If you are interested in creating some visualizations, then use your reporting tool of choice.

I used Microsoft Power BI as it can connect to cosmos DB and show data in real-time. Please see the below dashboard for your reference.

