# Azure Service Bus Mini Project

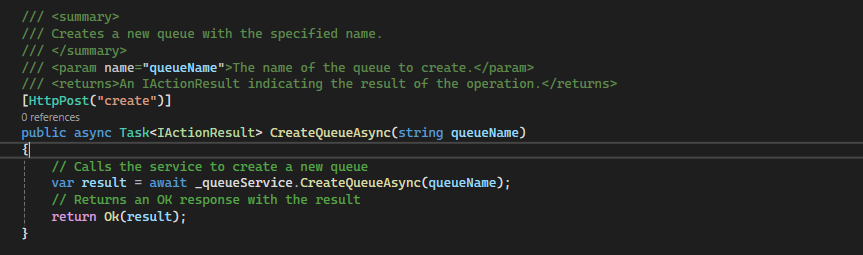


Figure 1

**Endpoint:** POST /api/queue/create  
**Description:**

* **Purpose:** Create a new queue with a specified name.
* **Key Rule:** Provide a unique queueName to avoid conflicts with existing queues.
* **Request Body:** The queueName should be passed as a query parameter.
* **Authorization:** Ensure the user is authorized to create queues.
* **Expected Result:** A success message with details about the newly created queue.
* **Sample Request:** POST /api/queue/create?queueName=myQueue
* **Considerations:** Validate the queueName to meet any required naming conventions or constraints.

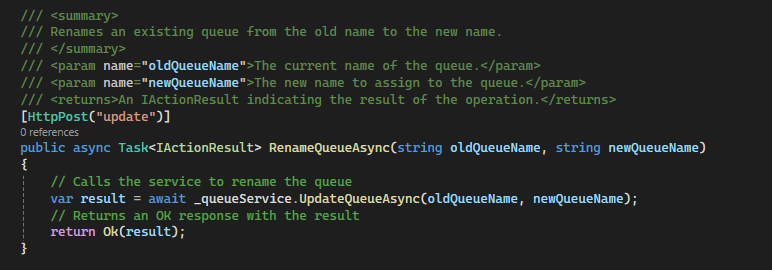


Figure 2

**Endpoint:** POST /api/queue/update  
**Description:**

* **Purpose:** Rename an existing queue from its old name to a new name.
* **Key Rule:** Ensure that the oldQueueName exists and the newQueueName is unique.
* **Request Body:** Provide both the oldQueueName and newQueueName as query parameters.
* **Authorization:** Ensure the user is authorized to rename queues.
* **Expected Result:** A success message with details about the renamed queue.
* **Sample Request:** POST /api/queue/update?oldQueueName=myOldQueue&newQueueName=myNewQueue
* **Considerations:** Handle cases where the oldQueueName does not exist or the newQueueName is already taken.

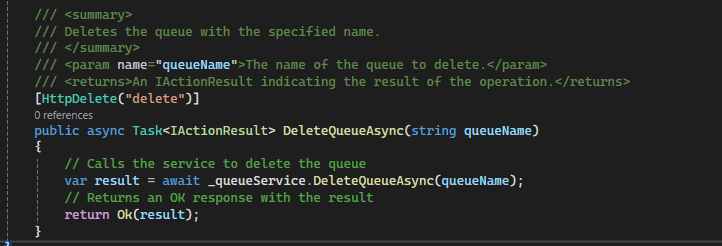


Figure 3

**Endpoint:** DELETE /api/queue/delete  
**Description:**

* **Purpose:** Delete a queue with the specified name.
* **Key Rule:** Provide the exact queueName of the queue you wish to delete.
* **Request Body:** The queueName should be passed as a query parameter.
* **Authorization:** Ensure the user is authorized to delete queues.
* **Expected Result:** A success message indicating that the queue was deleted.
* **Sample Request:** DELETE /api/queue/delete?queueName=myQueue
* **Considerations:** Confirm the queue is not in use or needed before deleting it to avoid unintended consequences.

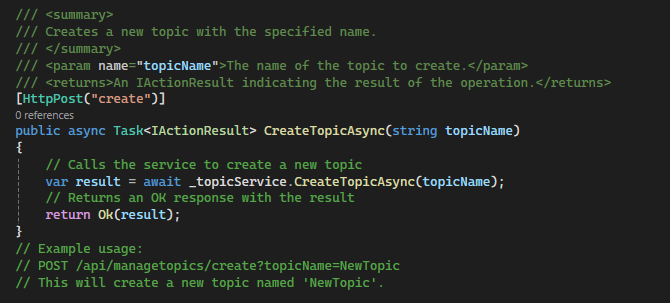


Figure 4

**Endpoint:** POST /api/managetopics/create  
**Description:**

* **Purpose:** Create a new topic with the specified name.
* **Key Rule:** Provide a unique topicName to avoid conflicts with existing topics.
* **Request Parameter:** topicName (string) - The name of the topic to create.
* **Authorization:** Ensure the user is authorized to create topics.
* **Expected Result:** A success message with details about the newly created topic.
* **Sample Request:** POST /api/managetopics/create?topicName=NewTopic
* **Considerations:** Validate the topicName to meet any required naming conventions or constraints.

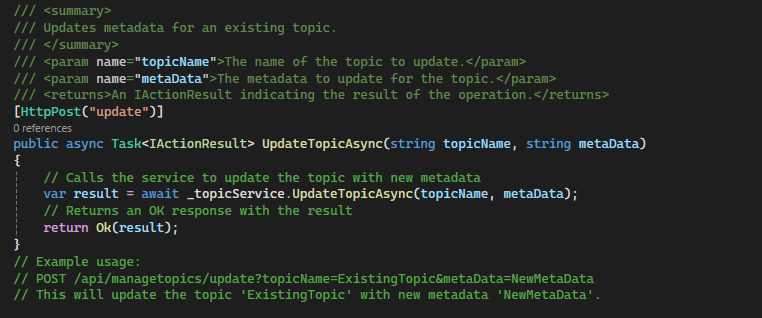


Figure 5

**Endpoint:** POST /api/managetopics/update  
**Description:**

* **Purpose:** Update the metadata for an existing topic.
* **Key Rule:** Ensure that the topicName exists before attempting to update its metadata.
* **Request Parameters:**
  + topicName (string) - The name of the topic to update.
  + metaData (string) - The new metadata to assign to the topic.
* **Authorization:** Ensure the user is authorized to update topic metadata.
* **Expected Result:** A success message with details about the updated metadata.
* **Sample Request:** POST /api/managetopics/update?topicName=ExistingTopic&metaData=NewMetaData
* **Considerations:** Handle cases where the topicName does not exist or if the metaData provided is invalid.

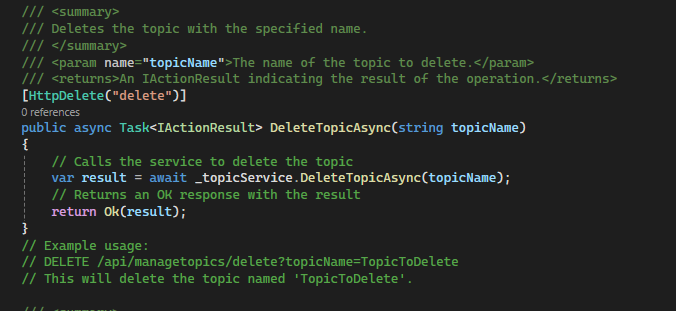


Figure 6

**Endpoint:** DELETE /api/managetopics/delete  
**Description:**

* **Purpose:** Delete a topic with the specified name.
* **Key Rule:** Provide the exact topicName of the topic you wish to delete.
* **Request Parameter:** topicName (string) - The name of the topic to delete.
* **Authorization:** Ensure the user is authorized to delete topics.
* **Expected Result:** A success message indicating that the topic was deleted.
* **Sample Request:** DELETE /api/managetopics/delete?topicName=TopicToDelete
* **Considerations:** Confirm the topic is not in use or needed before deleting it to avoid unintended consequences.

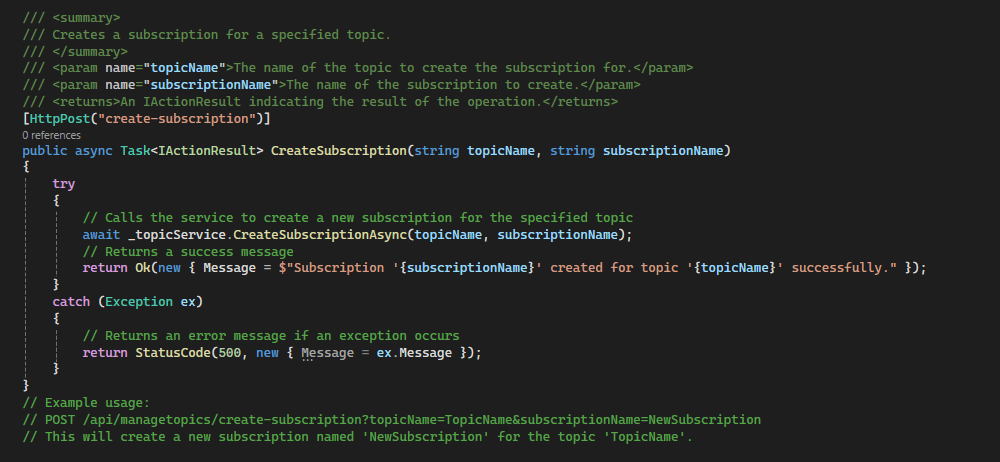


Figure 7

**Endpoint:** POST /api/managetopics/create-subscription  
**Description:**

* **Purpose:** Create a new subscription for a specified topic.
* **Key Rule:** Ensure that the topicName exists and the subscriptionName is unique.
* **Request Parameters:**
  + topicName (string) - The name of the topic for which to create the subscription.
  + subscriptionName (string) - The name of the subscription to create.
* **Authorization:** Ensure the user is authorized to create subscriptions for topics.
* **Expected Result:** A success message indicating that the subscription was successfully created.
* **Sample Request:** POST /api/managetopics/create-subscription?topicName=TopicName&subscriptionName=NewSubscription
* **Considerations:** Handle any exceptions that may arise, such as if the topicName does not exist or if the subscriptionName is already taken.

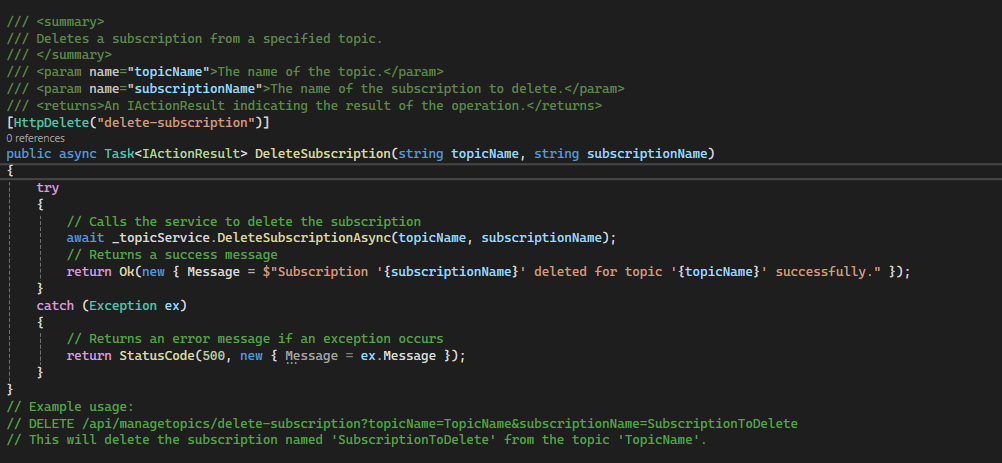


Figure 8

**Endpoint:** DELETE /api/managetopics/delete-subscription  
**Description:**

* **Purpose:** Delete a subscription from a specified topic.
* **Key Rule:** Ensure that the subscriptionName exists under the given topicName.
* **Request Parameters:**
  + topicName (string) - The name of the topic.
  + subscriptionName (string) - The name of the subscription to delete.
* **Authorization:** Ensure the user is authorized to delete subscriptions for topics.
* **Expected Result:** A success message indicating that the subscription was successfully deleted.
* **Sample Request:** DELETE /api/managetopics/delete-subscription?topicName=TopicName&subscriptionName=SubscriptionToDelete
* **Considerations:** Handle exceptions, such as if the subscriptionName does not exist.

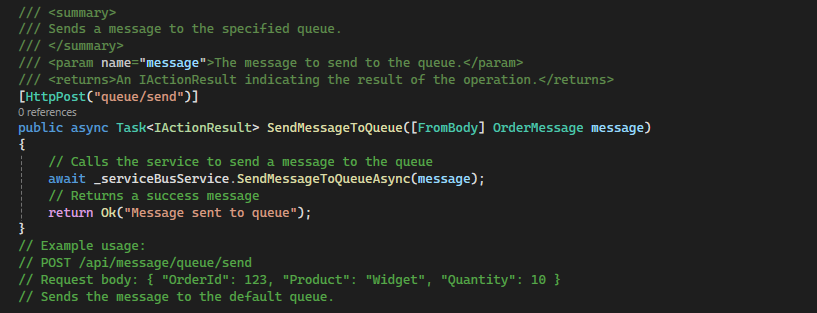


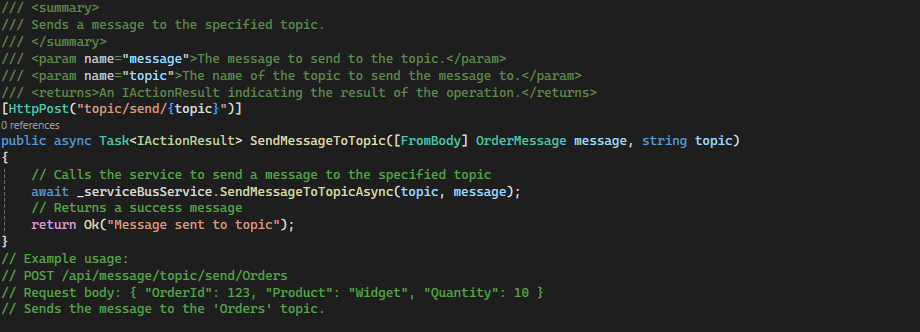
Figure 9

 **Description:** Sends a message to a specified queue.

 **HTTP Method:** POST

 **Endpoint:** /api/message/queue/send

 **Request Body:** JSON object representing the message (OrderMessage).



 **Description:** Sends a message to a specified topic.

 **HTTP Method:** POST

 **Endpoint:** /api/message/topic/send/{topic}

 **Parameters:** topic (Path parameter - the name of the topic).

 **Request Body:** JSON object representing the message (OrderMessage).

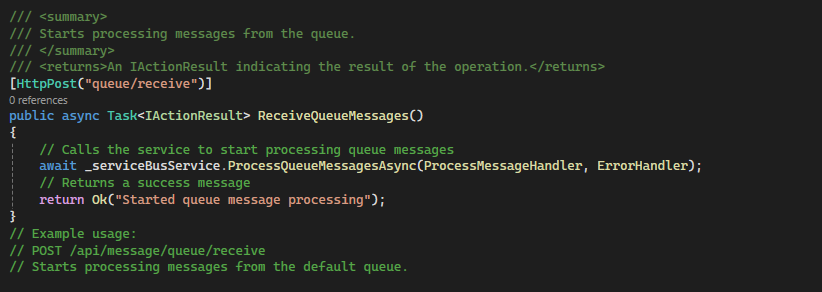


Figure 10

 **Description:** Receives a specified number of messages from a queue.

 **HTTP Method:** GET

 **Endpoint:** /api/message/receive-from-queue

 **Parameters:**

* queueName (Query parameter - the name of the queue).
* maxMessages (Query parameter - the maximum number of messages to receive).

 **Example Request:** /api/message/receive-from-queue?queueName=MyQueue&maxMessages=5

 **Response:** Returns the received messages from the specified queue.

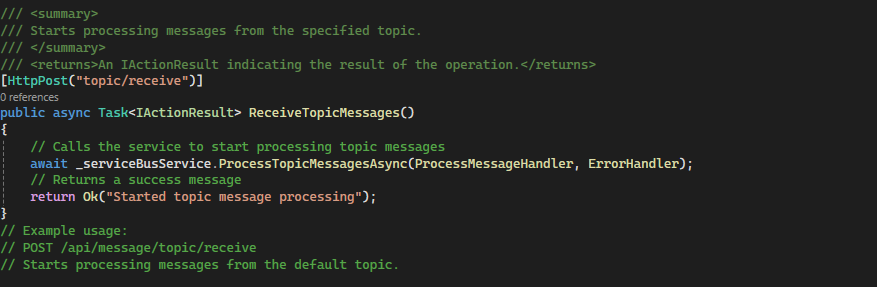


Figure 11

 **Description:** Receives a specified number of messages from a queue.

 **HTTP Method:** GET

 **Endpoint:** /api/message/receive-from-queue

 **Parameters:**

* queueName (Query parameter - the name of the queue).
* maxMessages (Query parameter - the maximum number of messages to receive).

 **Example Request:** /api/message/receive-from-queue?queueName=MyQueue&maxMessages=5

 **Response:** Returns the received messages from the specified queue.

## Example setup for azure service bus in azure

