Change Management - Setup a flow to automate the backup build a pipeline in ADO - document process.

Author: Nader Ghazala

Version: 1.0

**I- QnA Maker Knowledgebase backup:**

**Step1:** perform a dry-run using this link : [https://docs.microsoft.com/en-us/rest/api/cognitiveservices/qnamaker/knowledgebase/download](https://nam06.safelinks.protection.outlook.com/?url=https%3A%2F%2Furldefense.com%2Fv3%2F__https%3A%2Fdocs.microsoft.com%2Fen-us%2Frest%2Fapi%2Fcognitiveservices%2Fqnamaker%2Fknowledgebase%2Fdownload__%3B!!La4veWw!ntBRVCyjWdvrzAWn8nfBgA8kOameTuMPGOgQHhO5UrN25qAItLaUtl8C4preg5I7Hg%24&data=04%7C01%7CNader.Ghazala%40microsoft.com%7Cd8a648f9c5c7401ad9ad08d87cbfca11%7C72f988bf86f141af91ab2d7cd011db47%7C1%7C0%7C637396510476617806%7CUnknown%7CTWFpbGZsb3d8eyJWIjoiMC4wLjAwMDAiLCJQIjoiV2luMzIiLCJBTiI6Ik1haWwiLCJXVCI6Mn0%3D%7C1000&sdata=Dj6eP2ty4Z9WgC1Xt1X0B0byqTFeb8qBiIoU8SStVVY%3D&reserved=0)

**Step2:** Create a Logic App and a Storage Account from Azure portal to perform daily backup.

*Subscription* - **xxxxx**

*Resource Group* - **xxxxx**

All resources are to be created in **xxxx** region.

Chart

Description automatically generated

Graphical user interface, application

Description automatically generated

Your backup file should look like:

A picture containing scatter chart

Description automatically generated

**II- Knowledge Base automation for QnAMaker using Azure DevOps**

**Continuous Integration (**CI**) and Continuous Deployment (**CD**)**

Graphical user interface

Description automatically generated

Update using a json script which matches the QnAMaker REST API argument for[***Updating  KBs***](https://westus.dev.cognitive.microsoft.com/docs/services/5a93fcf85b4ccd136866eb37/operations/5ac266295b4ccd1554da7600)

{

"update": {

"qnaDocuments": [

{

"id": 1,

"answer": "I don’t have a body",

"source": "qna\_chitchat\_Professional.tsv",

"questions": {

"add": [ "Can you eat?" ]

}

}

]

}

}

*Update using a json script which just contains a URL for the questionnaire document.*

{

"update": {

"urls": [

"https://docs.microsoft.com/en-in/azure/cognitive-services/qnamaker/faqs",

"https://docs.microsoft.com/en-us/bot-framework/resources-bot-framework-faq"

]

}

}

Create Pipelines:

Graphical user interface, application, email

Description automatically generated

Get and Choose KB – Chooses a KB from the list of KBs in QnAMaker

# Write your PowerShell commands here.

Write-Host "Get and Chose KB";

$url = $(BaseURL) + "/knowledgebases"

$contentType = $(ContentType)

$headers =@{

$(ApiKey) = "$(Ocp-Apim-Subscription-Key)"

};

$response = Invoke-RestMethod -Method "GET" -Uri $url

-ContentType $contentType -Headers $headers

Write-Host $response;

$kbRef = $response.knowledgebases[0];

$knowledgeId = $kbRef.id;

Write-Host $knowledgeId;

"##vso[task.setvariable variable=kbId]$knowledgeId"

Update KB – Updates the chosen KB

# Write your PowerShell commands here.

Write-Host "Update KB";

$body = Get-Content $(artifactLocation)\UpdateKB.json;

Write-Host $body;

$url = $(BaseURL) + "/knowledgebases/$(kbId)"

$contentType = $(ContentType)

$headers =@{

$(ApiKey) = "$(Ocp-Apim-Subscription-Key)"

};

$response = Invoke-RestMethod -Method "PATCH" -Uri $url

-ContentType $contentType -Headers $headers -Body $body

Write-Host $response;

$operationId = $response.operationId;

"##vso[task.setvariable variable=opId]$operationId"

Check KB Update Status – Checks the status of the previous step

# Write your PowerShell commands here.

Write-Host "Check KB Update Status";

$url = $(BaseURL) + "/operations/$(opId)"

$contentType = $(ContentType)

$headers =@{

$(ApiKey) = "$(Ocp-Apim-Subscription-Key)"

};

$response = Invoke-RestMethod -Method "GET" -Uri $url

-ContentType $contentType -Headers $headers

Write-Host $response;

$operationState = $response.operationState;

Write-Host $operationState

$shouldContinue = $False;

if ($operationState -eq "Running")

{

Write-Host $operationState

$shouldContinue = $True;

}

"##vso[task.setvariable variable=opState]$shouldContinue"

Publish KB – Publishes a KB in QnAMaker

# Write your PowerShell commands here.

Write-Host "Publish KB";

Write-Host $(opState);

$url = $(BaseURL) + "/knowledgebases/$(kbId)"

$contentType = $(ContentType)

$headers =@{

$(ApiKey) = "$(Ocp-Apim-Subscription-Key)"

};

$response = Invoke-RestMethod -Method "POST" -Uri $url

-ContentType $contentType -Headers $headers;

Write-Host $response;