Function concept with PowerShell

Azure Function Apps is a "Function as a Service" solution and offers serverless execution of highly scalable code. This function can then be executed via HTTP requests and provide corresponding feedback in the HTTP response. However, this technology only makes sense if several requests are to be executed simultaneously (e.g. all devices are to fetch or save information). See chapter "Decision Support Technology (Comparison of Variants)".

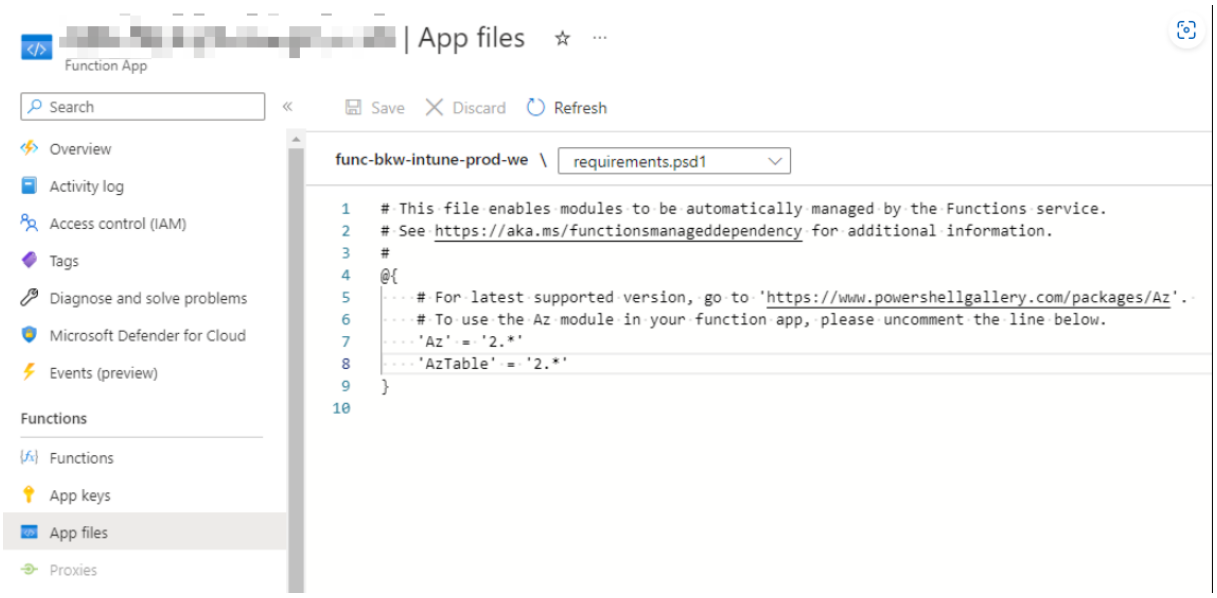
Function

The Functions are elements in the Azure Function Apps that contain the code and are then executed. Functions can have different triggers. Functions can be named according to the naming convention.

- FUNC-<Technology>-<KGTag>-<Object>-<Beschreibung>-<Status>-<Location>
  Example: FUNC-WIN-ALL-PS1-GetStorageTableContent-PROD-WE

PowerShell Modules

In order to use PowerShell modules, they must first be added to a config file in the Azure Function App. This can be achieved in the Function App via "App files" → "requirements.psd1" → Add module name and version to PowerShell list. However, adding the module to the Azure Function can take several 10 minutes.

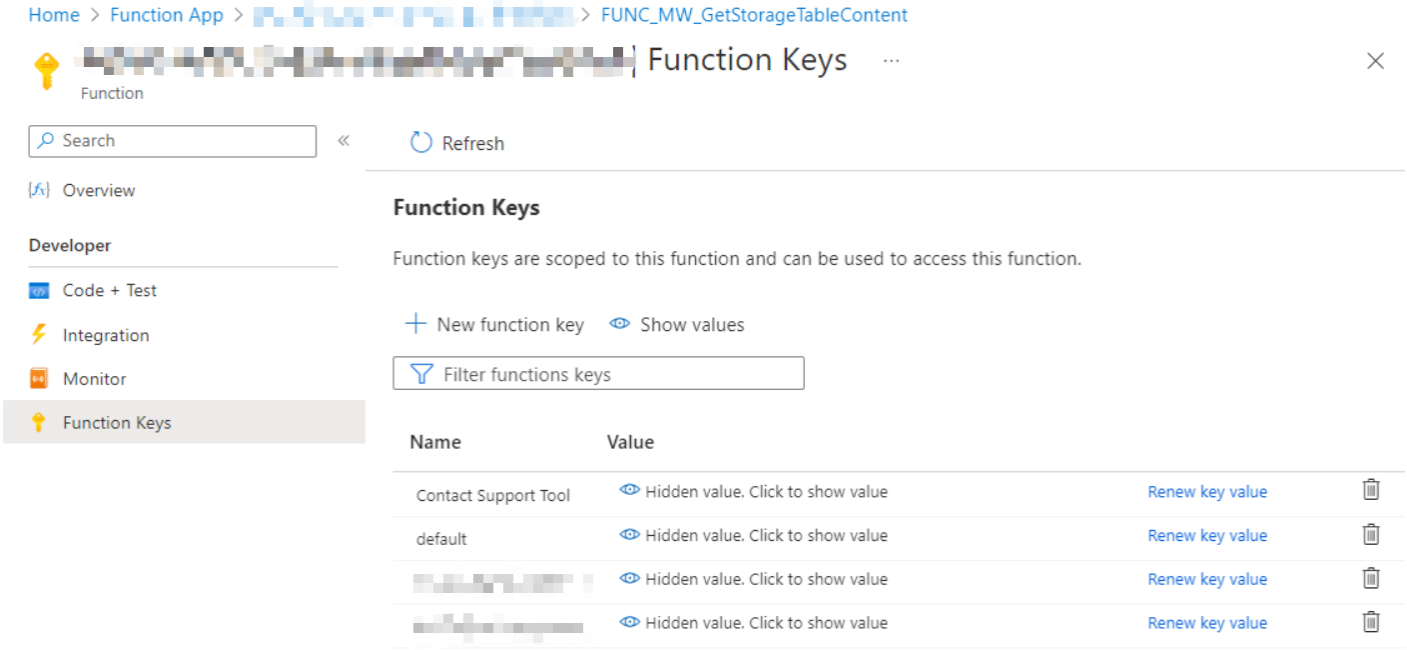


Import PowerShell Modules into scripts

```
Import-Module AzureAD
```

Authentication to Azure Function App

Authentication on the Azure function is done via the function keys, which are specified as a query. These function keys can be created on the function itself via → "Function Keys":



I don't use App Keys because I want to map multiple functions in an Azure Function App and then I wouldn't be able to use the security benefits of multiple keys.

The naming convention for Azure Function Keys is as follows:

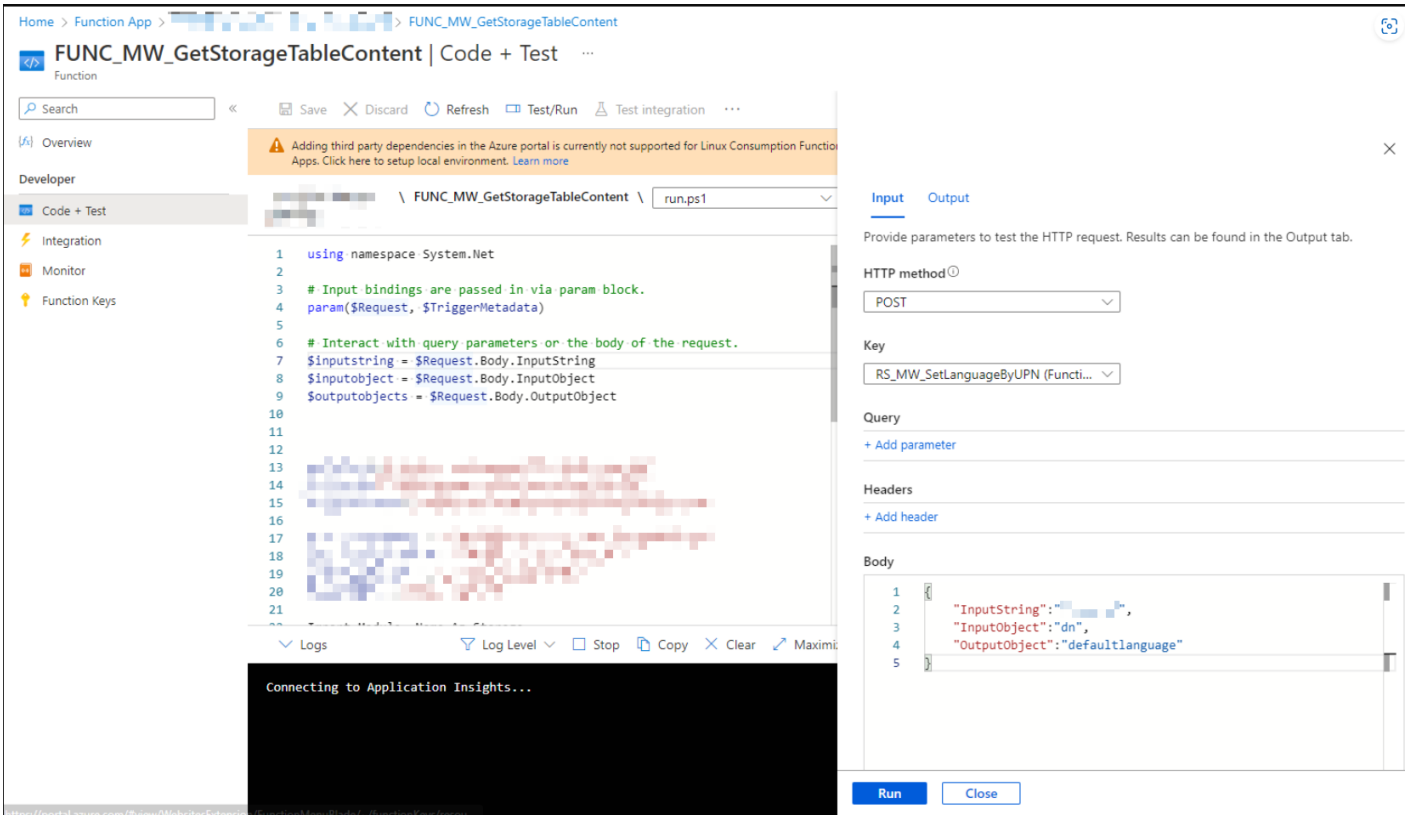
FKEY-<Technology>-<KGTag>-<Object>-<Description>-<Status>-<Location>.
Example: FKEY-RS-ALL-KEY-SetLanguageByUPN-PROD-WE

Triggers

Triggers mean types to call the Azure Function and execute the code.

Manual

The Azure Function can be started manually via the Azure Portal.

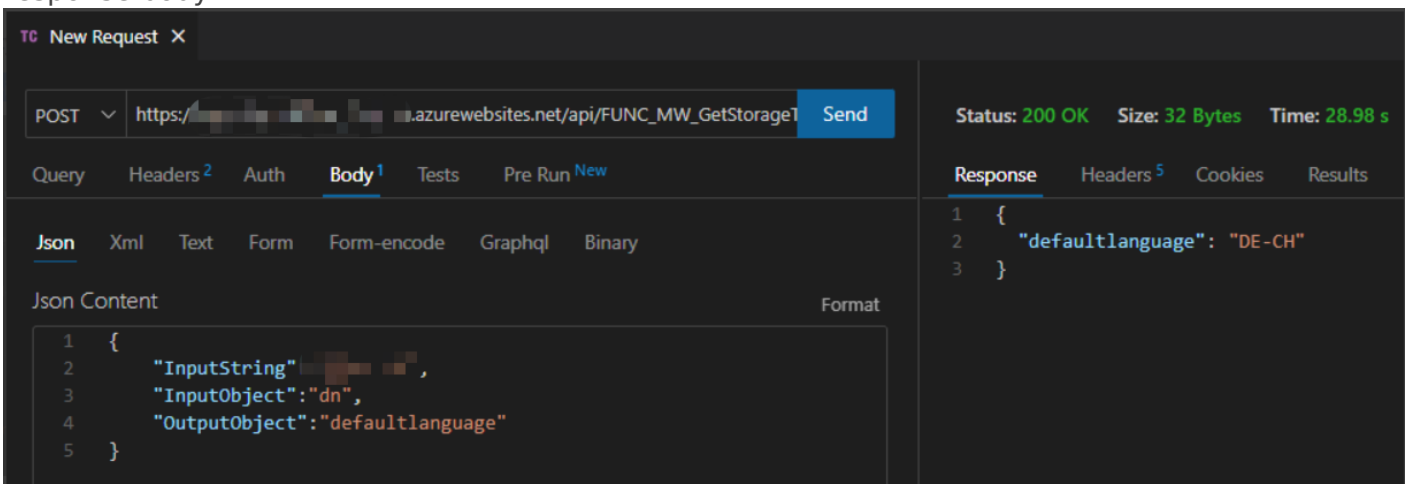


Schedules

Azure Functions can be executed regularly using Schedules. However, since Azure Runbook also offers this function, in most cases it makes more sense to create an Azure Runbook and not functions with schedules.

General REST Call

The function can be triggered via an HTTP request. The return can then also be found in the response body.



PowerShell

Via PowerShell, this can be achieved directly using the standard module "Invoke-Restmethod" with the corresponding URL and the function key.

```
$url="https://<azurefunctionappname>.azurewebsites.net/api/<azurefunction>?code=<azurefunctionkey>"
$Body = @"
{
   "InputString":"$TestVariable",
}
"@

$Output = (Invoke-Restmethod -uri $url -Body $Body -Method POST -ContentType "application/json").value
```

Revision #5
Created 1 December 2022 16:48:41 by Luca Noah Caprez
Updated 10 January 2023 07:39:50 by Luca Noah Caprez