Entwicklerdokumentation

## Anforderungen des Projektes

|  |  |
| --- | --- |
| **Nr.** | **Anforderungen** |
| F1 | Ausgabe als Notifikation |
| F2 | Ausgabe soll nur auf meinen Account auslösen |
| F3 | Ausgabe soll ist akkurat in Grad Celsius |
| F4 | Code überprüft ob es mit dem Internet verbunden ist |
| Q1 | Eingabefehler sollen abgefangen werden. |
| Q2 | Das Skript ist mit Kommentaren und einem Header versehen. |
| Q3 | Es existiert eine Entwicklerinformation und eine Betriebsdokumentation. |
| R1 | Das Skript muss innerhalb der vereinbarten Zeit fertiggestellt sein (inkl. Tests). |
| R2 | Das Skript muss in PowerShell (mind. Version 5.1) geschrieben sein. |
| R3 | Fremder Code und Fremdhilfe sind deklariert. |

## Pap des Ps1 Code

Ein Bild, das Text, Screenshot, Diagramm, Reihe enthält.

Automatisch generierte Beschreibung

## Code inkl. Kommentare und Header

<https://github.com/ascii-phoenix/Powershell-wetter-api/blob/main/Powershell-weather-api.ps1>

<#

.SYNOPSIS

    My code Gets the temperatur from a api and displays it with a notifacation

.DESCRIPTION

    My code checks if there is a specific user.

    After that, it checks if there is an internet connection with Google.

    If it succeeds, it makes an API call that gives me the temperature from a JSON file, which I display as a notification.

    If it fails 10 times, it gives a notification that there is no connection. If the API fails, there will be an error message.

.NOTES

Author: Eloi Knecht

    Build-version: 1.2

    Last updated : 2023-06-24

    Download BurntToast

    Header template was copied from scriptrunner.com

    Build in PS1-version 5.1.22621.3672

.COMPONENT

   Please downlode BurntToast and change the Usersting in the 42 line to yours.

.LINK

https://github.com/Windos/BurntToast.

https://open-meteo.com/

https://www.scriptrunner.com/en/blog/powershell-script-header-parameters-in-scriptrunner/

.Parameter user

 is the user

.Parameter connectedAdapter

 is a bool that checks that tere is a conection

.Parameter counterOfRepetition

Counts how many trys there where to connect with google

.Parameter apiUrl

 is a string of my API-URL

.Parameter response

 it is the API respons as a json file

.Parameter temperature\_2m

 is the temperture as a numbre

.Parameter imagePath

 is the image used in my nodification

#>

Import-Module BurntToast # (Install-Module -Name BurntToast) do not forget to downlode "BurntToast".

do {# Checks who is the user

    $user = (Get-WmiObject -Class Win32\_ComputerSystem).UserName

    if ($user -ne "") {

        if (-not($user -eq "Studio2Eloi\eloik")) {exit}

    } else {

        Start-Sleep -Seconds 5

    }

} while ($null -eq $user)

$counterOfRepetition = 0;

do { #Checks if there is a connection to google.com (only 10 trys)

    $connectedAdapter = Test-Connection -ComputerName google.com -Count 1 -Quiet

    $counterOfRepetition++

    if (!$connectedAdapter) {

        Start-Sleep -Seconds 5

    }

} while (!$connectedAdapter -and ($counterOfRepetition -lt 10))

if ($connectedAdapter -and $counterOfRepetition -lt 10) {

    $apiUrl = "https://api.open-meteo.com/v1/forecast?latitude=47.3925&longitude=8.0442&current=temperature\_2m,rain&forecast\_days=1"

# try was created with Chat GPT to prevent that $response is not $null and that the programm does not crash.

    try {# Try to make an api-call

        $response = Invoke-RestMethod -Uri $apiUrl -Method Get

        if ( $null -ne $response) {# cheks if there is an answer

            $temperature\_2m = $response.current.temperature\_2m

            $imagePath = "C:\Users\eloik\Powershell\sun-8553511\_1280.jpg"

            New-BurntToastNotification -Text "Heute ist es : $($temperature\_2m)C" -AppLogo $imagePath

        } else {

            $imagePath = "C:\Users\eloik\Powershell\sun-8553511\_1280.jpg"

            New-BurntToastNotification -Text "Leider keine API Verbindung" -AppLogo $imagePath

        }

    } catch {# if there is a Error

        $imagePath = "C:\Users\eloik\Powershell\sun-8553511\_1280.jpg"

        New-BurntToastNotification -Text "Leider gabe es ein Problem" -AppLogo $imagePath

    }

} else {

    $imagePath = "C:\Users\eloik\Powershell\sun-8553511\_1280.jpg"

    New-BurntToastNotification -Text "Leider keine API Verbindung" -AppLogo $imagePath

}

## Testfälle

|  |  |  |  |
| --- | --- | --- | --- |
| Nr. | Versuch Nummer | Erfolg? | Kommentar |
| F1 | F1.00 | true | BurntToast wurde benutzt |
| F2 | F2.00 | true | Beim Abbas hat es das Programm nicht ausgeführt duch das exit |
| F3 | F3.00 | true | Google sagt 21 c und mein code sagt 20.5 c |
| F4 | F4.00 | false | Falsches if Statement |
| F4 | F4.01 | true | If Statement zu ($counterOfRepetition -lt 10) verändert |
| Q1 | Q1.00 | true | Wenn der api nicht zurückgibt gibt es eine meldung |
| Q2 | Q2.00 | true | Glaube ein bisschen Overkill |
| Q3 | Q3.00 | true | Alles erstellt nach PPP |
| R1 | R1.00 | True? | ? |
| R2 | R2.00 | true | 5.1.22621.3672 |
| R3 | R3.00 | true | Alles, dass ich kopiert habe wurde deklariert |