
Modul 122

Leistungsbeurteilung LB1 - Dokumentation

Modul	IET-122 - Praxisarbeit
Eingereicht von	Levyn Schneider und David Meer Partnerarbeit
Eingereicht bei	Frau Yilmaz
Datum	21. Mai 2024

Änderungsverzeichnis

Datum	Version	Änderung	Autor
17.04.2023	0.5	Vorlage	Urs Dummermuth
23.04.2024	1	Dokumentation CLI E-Mail Tool	Levyn, David

Inhaltsverzeichnis

1	Ziele und Anforderungen	3
1.1	Einleitung	3
1.2	Zweck des Skriptes	3
1.3	Ziele	3
1.4	Anforderungen	3
2	Ablaufdiagramm	4
3	Skript/Programm (Realisierung)	5
3.1	Technologie	5
3.2	Ein- und Ausgabe	5
3.3	Kontrollstrukturen	5
4	Integration	6
4.1	Gmail E-Mail-Adresse	6
4.2	Gmail App Code	6
4.3	Dateien	6
4.4	Ausführung	6
5	Testfälle	7
5.1	Testfall	7
6	Reflexion	8
6.1	Journal	8
6.2	Fazit	9
7	Anhang	10
7.1	Skript	10

1 Ziele und Anforderungen

1.1 Einleitung

Levyn hat bereits viel Erfahrung in Bash und einiges an Erfahrung mit PowerShell. David hat sich ebenfalls mit PowerShell bereits viel beschäftigt. In seiner Firma hat er schon sämtliche PowerShell Scripts für diverse Automatisierungen erstellt.

1.2 Zweck des Skriptes

Wir erstellen eine CLI-E-Mail-Applikation, um einfach E-Mails im Terminal zu senden und empfangen. Dies haben wir uns ausfolgendem Grund überlegt; Falls einmal die E-Mail-Applikation nicht funktioniert, man aber trotzdem etwas empfangen oder auch senden möchte kann man dieses Tool verwenden.

1.3 Ziele

1. Bis Ende Projekt, kann man über das CLI-Tool E-Mails versenden.
2. Bis Ende Projekt, kann man über das CLI-Tool ungelesene E-Mails empfangen.
3. Bis Ende Projekt, werden ungelesene E-Mail in einem .txt speichern

1.4 Anforderungen

1.4.1 meine Anforderungen

- 1a: Es muss eine Sender-E-Mail-Adresse konfiguriert sein
- 1b: Es muss einen Gmail App Code konfiguriert sein.
- 1c: Es muss eine Empfänger-E-Mail-Adresse eingegeben werden.
- 1d: Es muss ein Betreff eingegeben werden.
- 1e: Es muss eine Nachricht eingegeben werden.

- 2a: Es muss eine Sender-E-Mail-Adresse konfiguriert sein.
- 2b: Es muss einen Gmail App Code konfiguriert sein.

1.4.2 Eingaben und Ausgaben

Eingaben:

- Absender-E-Mail-Adresse
- Gmail App-Code
- Empfänger-E-Mail-Adresse
- Betreff
- Nachricht
- Menu punkt Auswahlen

Ausgaben:

- Konfigurierte Sender-E-Mail-Adresse
- Konfigurierten Gmail App-Code
- Ungelesene E-Mails
- .txt Log-Dateien von den Empfangenen E-Mail.

1.4.3 Programmtechnische Anforderungen

1. Aktuellste PowerShell Version (7.4.2).
2. Eine Gmail E-Mail besitzen
3. Gmail App-Code erstellen

2 Ablaufdiagramm

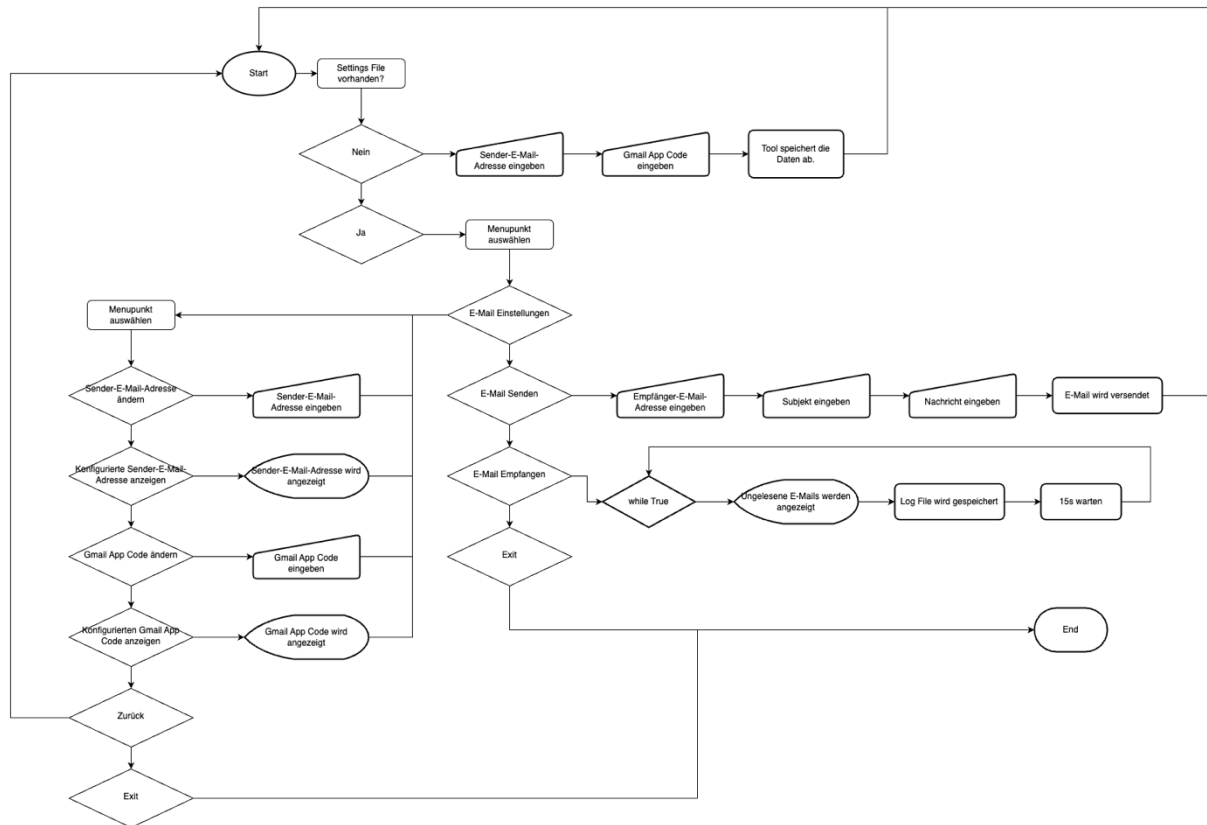


Abb. Ablaufdiagramm-CLI-E-Mail-Tool

3 Skript/Programm (Realisierung)

3.1 Technologie

PowerShell

3.2 Ein- und Ausgabe

```
$To = Read-Host "Gebe die E-Mail Adresse ein, an die die E-Mail gesendet werden soll"
```

```
$Subject = Read-Host "Gebe den Betreff der E-Mail ein"
```

```
$Content = Get-Content .\settings.txt
```

```
$From = $Content[0]
```

```
Write-Host "Ihre konfigurierte Sender-E-Mail-Adresse ist: $From"
```

3.3 Kontrollstrukturen

```
# Wiederholen, bis eine valide E-Mail eingegeben wurde
```

```
while ($null -eq $Email) {
```

```
    $Email = Read-Host "| Bitte geben Sie Ihre E-Mail Adresse ein"
```

```
    if ($Email -match "^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$" -and $Email -like  
    "**@gmail.com") {
```

```
        $Email | Out-File .\settings.txt
```

```
    $Password = $null
```

```
# Wiederholen, bis ein valides Passwort eingegeben wurde
```

```
while ($null -eq $Password) {
```

```
    $Password = Read-Host "| Bitte geben Sie Ihren Gmail App-Code ein"
```

```
    if ($Password -eq "") {
```

```
        Write-Host "|"
```

```
        Write-Host "| Der Gmail App-Code darf nicht leer sein"
```

```
        $Password = $null
```

```
        continue
```

```
    }
```

```
$Password | Out-File .\settings.txt -Append
Write-Host "|"
Write-Host "| E-Mail Adresse und Gmail App-Code erfolgreich konfiguriert"
Write-Host "|"
Write-Host "-----"
Write-Host
}
```

4 Integration

4.1 Gmail E-Mail-Adresse

Das Script unterstützt nur Gmail Adressen. Erstelle deshalb auf gmail.com ein Konto oder benutze ein bestehendes Google Konto.

4.2 Gmail App Code

Das Script basiert, zu der Gmail Adresse, noch auf einen Gmail App Code. Erstelle einen Gmail App Code über <https://myaccount.google.com/apppasswords> oder nutze einen bestehenden.

4.3 Dateien

Füge alle Dateien in einen gemeinsamen Ordner.

4.4 Ausführung

Führe die Datei `script.ps1` in deinem Ordner aus und folge den Anweisungen.

5 Testfälle

5.1 Testfall

NR	Anforderungen	Testfall	Erwartetes Resultat
01	Das `settings.txt` File ist nicht vorhanden	Das Programm sucht nach dem `settings.txt` File und findet es nicht.	Das Programm erstellt ein File und fragt den Nutzer nach E-Mail und Gmail App Code.
02	Settings.txt File mit korrekter Konfiguration	Das Programm sucht nach dem `settings.txt` File und prüft auf die Korrektheit.	Das Programm fährt normal fort.
03	-	Beim Menu-Selektor gibt der Nutzer eine ungültige Eingabe.	Das Programm stoppt.
04	Testfall 01	Das Programm fragt nach einer E-Mail-Adresse. Der Nutzer gibt eine nicht valide Adresse ein.	Das Programm wiederholt die Aufforderung, solange bis die E-Mail-Adresse valide ist.

6 Reflexion

6.1 Journal

Tätigkeit	Person
<ul style="list-style-type: none"> • Produktideen gesucht • Produktidee optimiert • Dokumentation grob abgeschlossen • Für die Programmierung informiert • Grundfunktionen entwickelt 	David
<ul style="list-style-type: none"> • Produktideen gesucht • Produktidee optimiert • Ablaufdiagramm • Für die Programmierung informiert • Grundfunktionen entwickelt 	Levyn
<ul style="list-style-type: none"> • Dokumentation Teil 1 abgeschlossen • Programm fast fertig entwickelt 	David
<ul style="list-style-type: none"> • Dokumentation Teil 1 abgeschlossen • Programm fast fertig entwickelt 	Levyn
<ul style="list-style-type: none"> • Programm fertig entwickelt und optimiert • Ablaufdiagramm optimiert 	Levyn
<ul style="list-style-type: none"> • Programm fertig entwickelt und optimiert 	David
<ul style="list-style-type: none"> • Dokumentation Teil 2 abgeschlossen 	David & Levyn

6.2 Fazit

6.2.1 Levyn

Ich fand das Projekt sehr spannend zum Entwickeln. Ich konnte einiges neues zu PowerShell und E-Mails lernen. Die Umsetzung des Projektes ist uns gut gelungen und wir konnten ein Script liefern, welcher auch ein wirklicher Nutzer hat. Mit David zu arbeiten war wie immer ziemlich angenehm. Wir hatten nie wirklichen Stress da wir so viel wie möglich vorgearbeitet haben und somit auch mehr leisten konnten. Ich bin gespannt auf das Python Projekt.

6.2.2 David

Das Projekt war ein voller Erfolg. Mit Levyn zusammen habe ich einiges über PowerShell dazugelernt. Insgesamt habe ich mir das ganze schwieriger vorgestellt, aber das Planen wie auch Umsetzen hat sehr gut funktioniert. Ebenso war unsere Zusammenarbeit sehr gut. Falls ich einmal etwas nicht begriff, hat mir Levyn es so erklärt, sodass ich einen guten Lehreffekt hatte und das Ganze dann problemlos selbst erledigen konnte. Wir waren uns meist einig und hatten keine grossen Komplikationen. Ich denke das Python-Projekt welches als nächstes folgt, wird ebenso ein voller Erfolg.

7 Anhang

7.1 Skript

Das Skript ist ungeeignet in der Dokumentation, wenn man das Skript einfach reinkopieren würde. Deswegen ist hier der GitHub Link: [ly-schneider/cli-email-tool \(github.com\)](https://github.com/ly-schneider/cli-email-tool)