|  |  |
| --- | --- |
|  | |
|  |  |
|  |  |
| Modul 122 | |
| Leistungsbeurteilung LB1 - Dokumentation | |
|  |  |
|  |  |
| Modul | IET-122 - Praxisarbeit |
| Eingereicht von | Leart, Lavan, Yannik, Leon |
| Datum | 18. Mai 2023 |

Änderungsverzeichnis

| Datum | Version | Änderung | Autor |
| --- | --- | --- | --- |
| 17.04.2023 | 0.5 | Vorlage | Urs |
| 08.05.2023 | 0.6 | Ziele & Einleitung | Leart |
| 08.05.2023 | 0.6 | Anforderungen | Leon |
| 08.05.2023 | 0.6 | Flussdiagramm | Lavan |
| 08.05.2023 | 0.6 | Scripten | Yannick |
| 15.05.2023 | 0.7 | Script nach Anforderungen anpassen | Yannick & Lavan |
| 15.05.2023 | 0.7 | Dokumentation Feinschliff | Leart |
| 18.05.2023 | 1.0 | Dokumentation Abschliessen | Lavan |

Inhaltsverzeichnis

[1 Ziele und Anforderungen 3](#_Toc132625387)

[1.1 Einleitung 3](#_Toc132625388)

[1.2 Zweck des Skriptes 3](#_Toc132625389)

[1.3 Ziele 3](#_Toc132625390)

[1.4 Anforderungen 3](#_Toc132625391)

[2 Ablaufdiagramm 4](#_Toc132625392)

[2.1 PAP / Flussdiagramm / Struktogramm 4](#_Toc132625393)

[2.2 Kommentar / Beschreibung 4](#_Toc132625394)

[3 Skript/Programm (Realisierung) 5](#_Toc132625395)

[3.1 Technologie Bash / Powershell / Python 5](#_Toc132625396)

[3.2 Ein- und Ausgabe 5](#_Toc132625397)

[3.3 Kontrollstrukturen 5](#_Toc132625398)

[4 Integration und Sicherheit 6](#_Toc132625399)

[4.1 Implementierung 6](#_Toc132625400)

[4.2 Sicherheit 6](#_Toc132625401)

[4.3 Kompatibilität 6](#_Toc132625402)

[4.4 Betrieb und Wartung 6](#_Toc132625403)

[5 Usecases und Testfälle 7](#_Toc132625404)

[5.1 Usecase 7](#_Toc132625405)

[5.2 Testfall 7](#_Toc132625406)

[6 Präsentation, Dokumentation 8](#_Toc132625407)

[6.1 Demo-Video 8](#_Toc132625408)

[7 Reflexion 9](#_Toc132625411)

[7.1 Journal 9](#_Toc132625412)

[7.2 Auswertung 10](#_Toc132625413)

[7.3 Fazit 10](#_Toc132625414)

Ziele und Anforderungen

## Einleitung

Unser Team besteht aus Lavan, Leart, Leon und Yannick. Beim Thema Scripting haben wir etwa alle die gleichen Vorkenntnisse. Unser Ziel in diesem Projekt ist es, uns so tief wie möglich uns in diesem Thema zu vertiefen, damit wir die Kenntnisse auch im Arbeitsalltag anwenden können. Bei den verschiedenen Aufgaben, die es im Projekt gibt, teilen wir uns auf aber trotzdem versuchen wir uns gegenseitig zu helfen und auch die Aufgaben der anderen zu verstehen.

## Zweck des Skriptes

Das PowerShell-Skript fordert den Benutzer dazu auf, den Pfad zu einem Ordner anzugeben, der komprimiert werden soll. Es prüft, ob der angegebene Pfad existiert. Anschließend wird der Benutzer aufgefordert, den Pfad zum Speicherort der ZIP-Datei anzugeben. Das Skript prüft erneut, ob der angegebene Pfad existiert. Es erstellt den Namen der ZIP-Datei basierend auf dem Namen des Ordners und prüft, ob bereits eine Datei mit demselben Namen existiert. Falls ja, wird der Benutzer gefragt, ob er die Datei umbenennen möchte. Bei Zustimmung generiert das Skript einen einzigartigen Namen für die ZIP-Datei. Bei Ablehnung bricht das Skript den Vorgang Ab. Sobald der Name der ZIP-Datei festgelegt ist, komprimiert das Skript den angegebenen Ordner in eine ZIP-Datei. Anschließend wird gezählt, wie viele Dateien komprimiert wurden und der Speicherort im Windows Explorer geöffnet. Das Skript fordert dann den Benutzer auf, seine E-Mail-Adresse einzugeben. Es sendet eine Bestätigungsnachricht an die angegebene E-Mail-Adresse,

in der es den Erfolg der Komprimierung bestätigt und die Anzahl der komprimierten Dateien angibt.

## Ziele

Das PowerShell-Skript erfüllt das Ziel der Benutzerinteraktion auf eine effiziente Weise. Es ist nicht einfach ein starrer Prozess, der im Hintergrund läuft, sondern es fordert aktiv Benutzereingaben und passt seine Aktionen entsprechend an. Indem es nach spezifischen Informationen fragt, wie den Pfad des zu komprimierenden Ordners, den Speicherort der ZIP-Datei und die E-Mail-Adresse des Benutzers, stellt das Skript sicher, dass es die genaue Aufgabe erfüllt, die der Benutzer erwartet.

Das Skript erfüllt auch das wichtige Ziel der Validierung. Es geht nicht einfach davon aus, dass die vom Benutzer bereitgestellten Informationen korrekt sind, sondern überprüft diese aktiv. Beispielsweise überprüft es, ob die vom Benutzer angegebenen Pfade tatsächlich existieren. Diese Art von Validierung trägt dazu bei, Fehler zu vermeiden und sicherzustellen, dass das Skript so reibungslos wie möglich läuft.

Ein weiteres wichtiges Ziel, das dieses Skript erfüllt, ist die Fehlervermeidung. Bevor es eine Datei erstellt, überprüft es, ob bereits eine Datei mit demselben Namen existiert. Auf diese Weise verhindert es, dass bestehende Dateien versehentlich überschrieben werden. Darüber hinaus bietet es dem Benutzer die Möglichkeit, einen einzigartigen Namen für die ZIP-Datei zu erstellen, wenn ein Namenskonflikt besteht. Dieses Merkmal zeigt eine sorgfältige Berücksichtigung möglicher Probleme und die Bereitstellung von Lösungen, um sie zu vermeiden.

Das Skript erfüllt das Ziel der Datenkomprimierung effektiv. Es nimmt den vom Benutzer angegebenen Ordner und komprimiert diesen in eine ZIP-Datei. Dies spart nicht nur Speicherplatz, sondern erleichtert auch den Transfer von Dateien. Die Fähigkeit, Dateien zu komprimieren, ist ein wesentlicher Aspekt der Datenspeicherung und -übertragung, und dieses Skript implementiert diese Funktion effizient.

Das Ziel der Benutzerinformierung wird auch gut erfüllt. Nach der Komprimierung zählt das Skript die Anzahl der Dateien im Ordner und teilt diese Information dem Benutzer mit. Diese Rückmeldung an den Benutzer ist wichtig, um ihm ein Verständnis dafür zu geben, was während des Prozesses passiert ist und wie effektiv die Komprimierung war.

Schließlich erleichtert das Skript die Benutzerführung, indem es den Speicherort im Windows Explorer öffnet. Dieses Merkmal ist besonders nützlich, da es dem Benutzer die manuelle Navigation zum Speicherort der ZIP-Datei erspart. Stattdessen können sie direkt auf die erstellte Datei zugreifen, sobald das Skript abgeschlossen ist. Dies zeigt erneut, wie das Skript den Prozess der Dateikomprimierung so einfach und benutzerfreundlich wie möglich gestaltet.

## Anforderungen

### Allgemeine Anforderungen

Benutzerfreundlichkeit: Das Skript sollte einfach zu bedienen und leicht verständlich sein.

Effizienz: Das Skript sollte den Komprimierungsvorgang schnell und mit minimaler Beeinträchtigung der Systemleistung ausführen.

Automatisierung: Das Skript sollte in der Lage sein, den Komprimierungsprozess und den E-Mail-Versand mit minimaler Benutzerinteraktion durchzuführen.

### Eingaben und Ausgaben

Eingaben: Der Benutzer sollte in der Lage sein, den Pfad des zu komprimierenden Ordners, den Speicherort der ZIP-Datei und seine E-Mail-Adresse einzugeben.

Ausgaben: Eine ZIP-Datei des komprimierten Ordners und eine Bestätigungsnachricht per E-Mail.

### Programmtechnische Anforderungen

Fehlerprüfung: Das Skript sollte überprüfen, ob die eingegebenen Pfade gültig sind und existieren.

E-Mail-Integration: Das Skript sollte eine Bestätigungsnachricht per E-Mail senden können, wenn der Komprimierungsvorgang abgeschlossen ist.

# Ablaufdiagramm

## PAP / Flussdiagramm / Struktogramm

Flussdiagramm als Beilage.

## Kommentar / Beschreibung

Das Ablaufdiagramm zeigt den Prozess, der vom Skript ausgeführt wird. Es beginnt mit der Eingabeaufforderung für den Ordnerpfad, gefolgt von der Überprüfung des Pfades und der Eingabeaufforderung für den Speicherort der ZIP-Datei. Anschließend wird der Pfad erneut überprüft, bevor der Komprimierungsvorgang beginnt. Schließlich wird eine Bestätigungsnachricht per E-Mail gesendet.

# Skript/Programm (Realisierung)

## Technologie Bash / Powershell / Python

Powershell Version 3.0+

## Ein- und Ausgabe

Eingabeaufforderungen für Ordnerpfad, Speicherort der ZIP-Datei und E-Mail-Adresse des Benutzers.

Ausgabe der komprimierten ZIP-Datei und Bestätigungsnachricht per E-Mail.

## Kontrollstrukturen

Bedingte Anweisungen zur Überprüfung der Gültigkeit der eingegebenen Pfade.

Schleifen für Fehlerbehebung und Wiederholung von Eingabeaufforderungen bei Bedarf.

# Integration und Sicherheit

## Implementierung

Das Skript sollte auf Windows-Systemen lauffähig sein und in die bestehende PowerShell-Umgebung integriert werden können.

## Sicherheit

Interaktion sollte so lange wie möglich Lokal verlaufen. Programm sollte so aufgebaut sein, dass man den Code nur bei der Einrichtung anpassen muss.

## Kompatibilität

Das Skript sollte mit den gängigen Windows-Versionen und -Umgebungen kompatibel sein.

## Betrieb und Wartung

Das Skript sollte leicht zu warten und zu aktualisieren sein, um zukünftige Anforderungen und Änderungen zu berücksichtigen. Jedoch ist die Langlebigkeit einer der Hauptziele.

# Usecases und Testfälle

## Usecase

* Automatisierte Komprimierung eines angegebenen Ordners.
* Aussortierung und Strukturierung von Backup Daten.
* Flexibilität für den Speicherort der Backup Daten.

## Testfall

Es wurden alles Ablaufmöglichkeiten geprüft und optimiert, damit wir die beste Nutzererfahrung versichern können. Für jeden möglichen Fall haben wir Beschreibungen drinnen, welche dem Endbenutzer erklären, warum es zu diesem Fall geführt hat. Als Abschluss wurde unser Script, durch externe Personen auf die Funktionalität geprüft und abgesegnet.

* Benutzer hinterlegt einen falschen Ordner, welcher Komprimiert werden soll -> dem Benutzer wird übermittelt, dass es keinen Ordner gibt mit dem angegebenen Pfad.
* Benutzer hinterlegt einen falschen Ordner, welcher bei dem das komprimierte File abgespeichert werden soll -> dem Benutzer wird übermittelt, dass es keinen Ziel-Ordner gibt mit dem angegebenen Pfad.
* Benutzer hinterlegt keine E-Mail Adresse als Bestätigung -> der Vorgang wird abgebrochen.

# Präsentation, Dokumentation

## Demo-Video

Als Beilage mit dabei.

# Reflexion

## Journal

| **Tag** | **Tätigkeit** | **Person** | **Bemerkungen** |
| --- | --- | --- | --- |
| 1 | Idee finden, Brainstorming | TEAM |  |
| 2 | Dokumentation Einleitung & Ziele  Flussdiagramm erstellen und Gruppenverteilung  PowerShell-Skript erstellt, das einen angegebenen Ordner in eine ZIP-Datei komprimiert und die ZIP-Datei umbenennt, wenn eine Datei mit demselben Namen bereits vorhanden ist. Das Skript sendet auch eine Bestätigungsnachricht per E-Mail an den Benutzer. Ich war sehr zufrieden damit, wie effektiv und robust meine Lösung war.  Heute habe ich an der Dokumentation gearbeitet. | LR  LA  YA  LG | Ich habe die Ziele und die Einleitung unseres Projekts geschrieben das wir als Team immer eine Orientierung haben, wo wir hinarbeiten möchten. Das nächste Mal muss ich die Doku nach dem Beurteilungsraster noch anpassen.  Das nächste Mal muss der Code, nach den Excel Vorgabe angepasst werden und Flussdiagramm updaten.    Ich habe zu lange an dem Punkt gebraucht fürs Finden des Unique-Passworts der Google API  Das nächste Mal könnte ich etwas schneller und kreativer schreiben, so dass wir besser vorankommen. |
| 3 | To-Dos erstellt für den Entwickler und für den Protokollführer.  Ich habe als Entwickler den Code nach den To-Dos von Lavan und den Vorgaben der Excel-Datei angepasst  Dokumentation nach Bewertungsraster angepasst | LA  YA  LR | Das nächste Mal steht nur die Projektabgabe vor.  Projektabgabe + Video  Ich habe heute die Ziele mit SMART formuliert. Die Doku habe ich noch schön formatiert und bei den |

## Auswertung

Es war eine Gelegenheit, unsere technischen Fähigkeiten zu nutzen und zu erweitern, sondern auch eine Chance uns als Team weiterzuentwickeln. Wir begannen dieses Projekt mit einer klaren Vision, dass technisch anspruchsvoll, benutzerfreundlich und effizient ist.

Die Erfahrung hat uns nicht nur in unseren technischen Fähigkeiten gestärkt, sondern uns auch wertvolle Lektionen in Bezug auf Teamarbeit, Kommunikation und Projektmanagement vermittelt. Das fertige Skript ist nicht nur ein Beweis unserer harten Arbeit und unseres Engagements, sondern auch ein Zeugnis für unser Lernen und Wachstum.

## Fazit

Abschließend können wir sagen, dass dieses Projekt eine bereichernde Erfahrung war, die uns sowohl auf individueller als auch auf Team-Ebene gestärkt hat. Wir freuen uns darauf, diese Lektionen in zukünftigen Projekten anzuwenden und unser kontinuierliches Wachstum und unsere Weiterentwicklung durch unsere Lehre fortzusetzen.