



M143 ST18A LB3



Luis Lüscher

Inhalt

1. Vorwort.....	2
2. Vorgehensweise	3
2.1. Konzept.....	3
2.2. Ausgangslage	3
2.3. Idee.....	3
2.4. Zeitplanung.....	3
3. Umsetzung.....	4
3.1. Konzept.....	4
3.2. Vorgehen	4
3.3. Skript Erklärung	5
4. Sicheres GUI, testen der Anwendung.....	6
Testgruppe <<A>>	6
Testgruppe <>.....	6
Testgruppe <<C>>.....	6
5. Hilfestellung.....	7
6. Reflexion.....	7
7. Schlusswort	7
8. Quellenangabe	7

1. Vorwort

Ich bin Mitarbeiter des IT-Unternehmen «Redundant Data Corporation». Ich werde nun für die Yeez AG ein Datensicherungskonzept realisieren. Ich werde die Arbeit mit möglichst viel Kreativität angehen. Ich möchte ein Programm entwickeln welches Daten sowie Unterordner von Punkt A zum Punkt B bringt. Dabei soll es ein GUI haben, welches für den User einfach zu bedienen ist. Ein restore sollte ebenfalls möglich sein.

2. Vorgehensweise

2.1. Konzept

Ich möchte ein GUI erstellen welches dem User es leicht ermöglicht Daten zu backupen und restoren. Dabei sollte im Hintergrund Robocopy laufen.

2.2. Ausgangslage

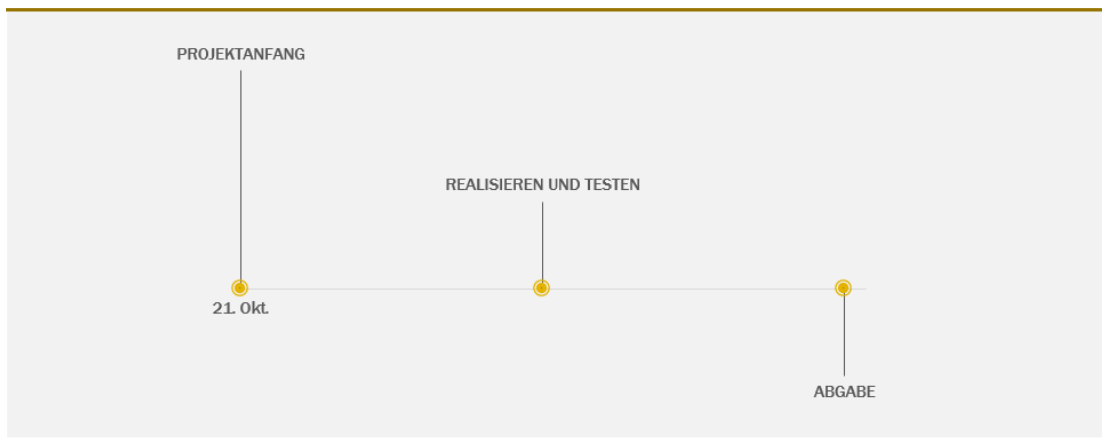
Ich habe in der LB2 nun die notwendigen Systeme aufgesetzt und konfiguriert. Durch die LB3 möchte ich dem Kunden ermöglichen seine Daten selbst zu backupen bzw. restoren. Ich bin nun mit meiner Firma Redundant Data Corporation dran, dieses Projekt zu realisieren.

2.3. Idee

Ein einfaches und übersichtliches Skript welches, Voll-, Incremental- und Differential Backup sowie restore beinhaltet. Der User sollte durch ein Handbuch auch ohne IT-Kenntnisse das Skript ausführen können. Das GUI werde ich mit Primalforms erstellen.

2.4. Zeitplanung

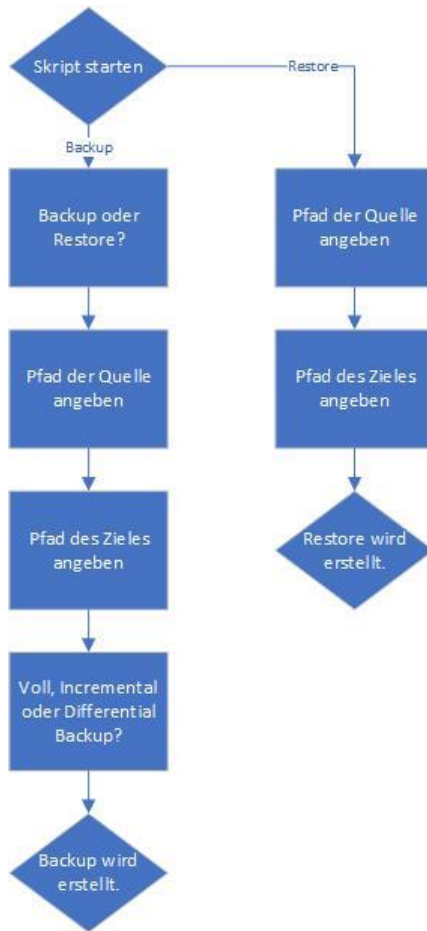
PROJEKTZEITACHSE



Datum	21.10.2019	28.10.2019	04.10.2019
Tätigkeit			
Vorbereitung	X		
Realisieren		X	
Testen		X	
Abgabe			X

3. Umsetzung

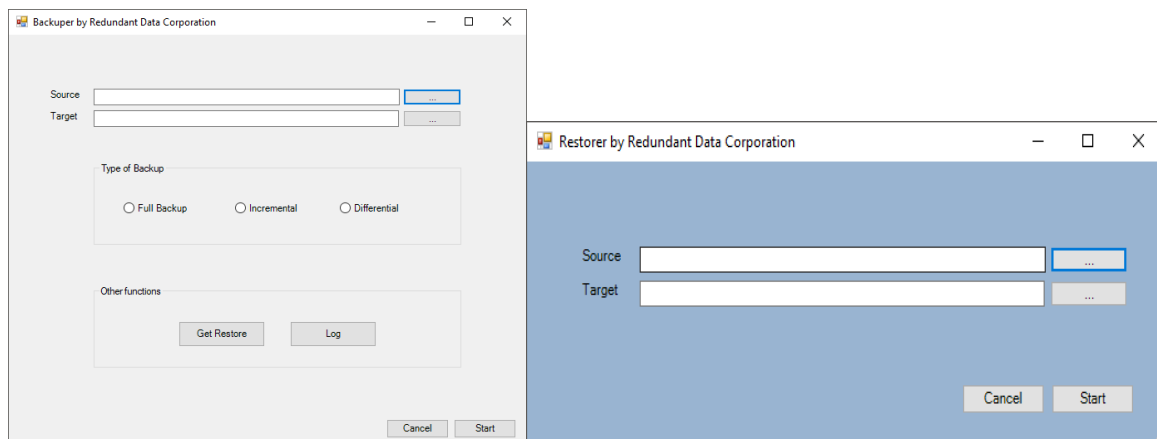
3.1. Konzept



Der User sollte die Möglichkeit haben, dass er entweder Restoren oder Backups kann. Danach sind die Prozeduren gleich. Alle Schritte werden in einem Log file dokumentiert.

3.2. Vorgehen

Zuerst habe ich mir Gedanken gemacht, wie ich genau vorgehen soll. Ich begann dann mit der Programmprozedur, also wie das Skript ablaufen soll und habe dies dann in Visio visualisiert. Danach habe zwei GUI in Primal Forms erstellt eines für das Backupen und eines für das Restoren. Danach ging es an das Skripten, da ich bereits Robycopy kannte war es für mich keine grosse Herausforderung. Anschliessend habe ich diese angefangen diese Dokumentation zu schreiben.



3.3. Skript Erklärung

Log function

Ich habe zuerst eine eigene Log function geschrieben, die folgendermassen aussieht:

```
$path = "\\192.168.0.100\home\TBZ\M143\LB3"
$date = get-date -format "yyyy-MM-dd"
$file = ("Log_" + $date + ".log")
$logfile = $path + "\" + $file

function write-Log([string]$logtext, [int]$level = 0) {
    $logdate = get-date -format "yyyy-MM-dd HH:mm:ss"
    if ($level -eq 0) {
        $logtext = "[INFO] " + $logtext
        $text = "[" + $logdate + "]" - " + $logtext
    }
    $text >> $logfile
}
write-Log "Starting the application"
```

Durch die function « write-log» kann ich dnn im Script etwas in ein Log file füllen.

Öffnen des log files

```
write-log "Opening Log file"
$path = "\\192.168.0.100\home\TBZ\M143\LB3"
$date = get-date -format "yyyy-MM-dd"
$file = ("Log_" + $date + ".log")
$logfile = $path + "\" + $file
```

```
Invoke-Item $logfile
```

Durch Invoke-Item wird das Logfile geöffnet

Öffnen des Explorer für Ordner Auswahl

```
Add-Type -AssemblyName System.Windows.Forms
$FolderBrowser_target = New-Object System.Windows.Forms.FolderBrowserDialog
[void]$FolderBrowser_target.ShowDialog()
$FolderBrowser_target.SelectedPath

$textbox2.text = $FolderBrowser_target.SelectedPath
$global:target = $textbox2.Text
write-log "Selected target is visible"
```

Öffnen des Explorer via New-Object zudem wird der Pfad in der Textbox angezeigt.

Welcher Radiobutton ist ausgewählt?

```
if($radioButton1.Checked -eq $true)
```

Durch if-Statements wird überprüft welcher RadioButton ausgewählt ist.

Schliessen des Forms nach Tätigkeit

```
$form1.close()
```


Sicheres GUI Fehlermeldung bei fehlenden Daten

```
if (($textBox1.TextLength -eq 0) -or ($textBox2.TextLength -eq 0)){
[System.Windows.Forms.MessageBox]::Show("You must select source and target" ,
"ERROR")
write-log "There are not all data"
```


Wenn die Textbox 1 oder 2 keinen Wert haben, kommt ein Pop-Up.

4. Sicheres GUI, testen der Anwendung


Testgruppe <<A>>

Beschreibung Testgruppe: Wird bei drei verschiedenen Backup Arten unterschieden?				
Setup:				
#	Nötige Eingabe/Aktion	Erwartete Ausgabe/Aktion	Tatsächliche Ausgabe/Aktion	
A1	RadioButtons jeweils benutzen	Daten werden richtig verarbeitet	<i>Daten werden richtig verarbeitet</i>	OK
A2	Robocopy Befehle sind richtig	Daten werden richtig verarbeitet	Alle Daten werden richtig übernommen	OK

Testgruppe <>

Beschreibung Testgruppe: Ist ein Restore möglich?				
Setup:				
#	Nötige Eingabe/Aktion	Erwartete Ausgabe/Aktion	Tatsächliche Ausgabe/Aktion	
B1	Restore Button vorhanden bzw. funktioniert	Button vorhanden + funktioniert	<i>Ist ok</i>	OK
B2	Robocopy restore funktioniert	Befehle funktionieren	Ist OK	OK

Testgruppe <<C>>

Beschreibung Testgruppe: Werden Logs gemacht?				
Setup:				
#	Nötige Eingabe/Aktion	Erwartete Ausgabe/Aktion	Tatsächliche Ausgabe/Aktion	
C1	Logs einsehen	Logs sind einsehbar für den User	<i>Ist ok</i>	OK
C2	Logs werden für wichtige Schritte erstellt	Logs sind einsehbar	Ist OK	OK

5. Hilfestellung

Ich selbst habe ein wenig Hilfe von Chris Zimmermann bekommen, da ich am Anfang nicht ganz verstanden habe wie ich genau beginnen sollte. Ansonsten habe ich Seymen bei einigen Commands in Powershell geholfen.

6. Reflexion

21.09.2019:

Heute habe ich einige Vorbereitungen gemacht zudem habe ich mich intensiver mit Robocopy befasst, da mich das Thema sehr interessiert hat.

28.09.2019:

Heute habe das GUI mit Primalforms erstellt und einige Codezeilen bereits gemacht. Zudem habe ich mit meiner Dokumentation angefangen

04.10.2019:

Heute habe ich meine Dokumentation vollendet und nochmals mein Script getestet es funktioniert ohne Probleme.

7. Schlusswort

Ich bin sehr zufrieden mit meinem Script. Es war nicht besonders schwer, die Arbeit zu erledigen. Sie bereitete mir viel Spass und ich konnte mein bereits vorhandenes Know-How gut anwenden, was mich natürlich sehr stolz machte.

8. Quellenangabe

Beispiel Script von Marcello Calisto
Ryan Simmonds (Hilfe im Betrieb)