Obligatorisches Deckblatt für den Bericht, unter transparenter Abdeckfolie,
z.H. IPA-verantwortlicher Fachkraft für Korrektur (gelb)

IPA Informatiker/in 2018

Schöpfer Patrick

Facharbeit

Neue Funktionen Testautomationstool TsNet V2

Änderungen von Hand nachtragen

1

Kandidat/Kandidatin
Schöpfer Patrick
Wylen-Bantlipark 11
6440 Ingenbohl

P 0799476139
G +41 41 724 51 95
patrick.schoepfer@siemens.com

BerufsbildnerIn/Lehrfirma
Häusler Martin
Siemens Schweiz AG
Gubelstrasse 22
6301 Zug

+41 58 724 24 24
martin.r.haeusler@siemens.com

IPA-verantwortliche Fachkraft
Speckien Michael
Siemens Schweiz AG
Gubelstrasse 22
6300 Zug

+41 58 7245556
michael.speckien@siemens.com

"Dienstweg" Exemplar Nr 1:

1. IPA-verantwortlicher Fachkraft zur Korrektur

2. Besprechung der Korrektur nach dem Fachgespräch

3. Hauptexperte schickt diesen Bericht an den Chefexperten z.H. der Notenkonferenz

4. Der Bericht wird nach der Einsprachefrist vernichtet.

Mit ihrer Unterschrift bestätigt **die IPA-verantwortliche Fachkraft** auch, dass sie das Summary auf PkOrg geprüft und für die Veröffentlichung freigegeben hat und keine Rechte Dritter verletzt werden.	IPA-verantwortliche Fachkraft Korrigiert (Datum, Unterschrift)
Mit seiner Unterschrift bestätigt **der/die Hauptexpert/in**, dass sie Korrekturen und Hinweise der IPA-verantwortlichen Fachkraft in diesem Bericht zur Kenntnis genommen hat.	Hauptexperte Visiert (Datum, Unterschrift)
Der Kandidat/die Kandidatin bestätigt, dass er die vorliegende Facharbeit während der im Journal deklarierten Zeit selbstständig und ohne fremde Hilfe ausgeführt und keine anderen als die angegebenen Hilfsmittel verwendet hat. Insbesondere sind alle Übernahmen von Texten und Programmen aus anderen Quellen als solche gekennzeichnet.	Kandidat / Kandidatin Bestätigung (Datum, Unterschrift)

Inhalt

1. Teil 1 5
1.1 Einführung 5
1.1.1 IPA 5
1.1.2 Zweck des Dokumentes 5
1.1.3 Ziel Publikum 5
1.2 Projektauftrag gemäss PkOrg 6
1.2.1 Ausgangslage 6
1.2.2 Detaillierte Aufgabenstellung 7
1.2.3 Mittel und Methoden 7
1.2.4 Vorkenntnisse 8
1.2.5 Vorarbeiten 8
1.2.6 Neue Lerninhalte 8
1.2.7 Arbeiten in den letzten 6 Monaten 8
1.3 Projektorganisation 9
1.3.1 Arbeitsplatz 9
1.3.2 Versionierung 9
1.3.3 Personen und Dienste 10
1.3.4 Abteilung BT CPS &D ZG CS SAP 10
1.3.5 Projektmanagement Methode 11
1.4 Planung 12
1.4.1 Soll-Zeitplan 12
1.4.2 Soll-Ist-Vergleich 13
1.4.3 Tätigkeiten 14
1.4.4 Meilensteine 16
1.4.5 Visualisierung der Arbeitsschritte 17
1.4.6 Erklärung Flussdiagramm 20
1.5 Arbeitsjournal 20
1.5.1 Zweck 20
1.5.2 Anwendungsbereich 20
1.5.3 Aufbau/Legende 20
1.5.4 Arbeitsjournale 22
2. Teil 2 32

2.1 IPA Kurzfassung 32
2.1.1 Ausgangssituation 32
2.1.2 Umsetzung 32
2.1.3 Ergebnis 32
2.2 Realisierung 33
2.2.1 Umfeld 33
2.2.2 Bedienoberfläche / UI 35
2.2.3 Hilfstabelle 40
2.2.4 Diagramm 42
2.2.5 Funktionen 43
2.2.6 Felder 50
2.2.7 Entwicklungsumgebung 51
2.3 Kontrollieren 52
2.3.1 Test Konzept 52
2.3.2 Ergebnis 60
2.3.3 Naschtests 61
3. Auswerten 63
3.1 Schlussbericht 63
4. Glossar 64
5. Quellen 65
6. Anhang 66

Tabellen Verzeichnis

Tabelle 1: Projektauftrag 6
Tabelle 2: Personen und Dienste 10
Tabelle 3: Tätigkeiten Beschreibung 16
Tabelle 4: Meilensteine 16
Tabelle 5: Flussdiagramm Erklärung 20
Tabelle 6: Arbeitsjournal allgemeine Informationen 20
Tabelle 7: Arbeitsjournal Informationen zu den Tätigkeiten 21
Tabelle 8: Interpolierung 48
Tabelle 9: Interpolierung Formel 48
Tabelle 10: Range anzeige 49
Tabelle 11: Formel False Value 50
Tabelle 12: Formel True Value 50
Tabelle 13: Felder 50
Tabelle 14: Glossar 65

Abbildungsverzeichnis

Abbildung 1: Arbeitsplatz 9
Abbildung 2: Ordnerstruktur Server 9
Abbildung 3: Lokale Backup Ordnerstruktur 10
Abbildung 4: IPERKA 11
Abbildung 5: Soll Zeitplan 12
Abbildung 6: Soll-Ist-Zeitplan 13
Abbildung 7: Flussdiagramm Trend Button 17
Abbildung 8: Flussdiagramm Create Trend Teil1 18
Abbildung 9: Flussdiagramm Create Trend Teil2 19
Abbildung 10: TsNet Test 33
Abbildung 11: TsNet Overview 34
Abbildung 12: User Interface 35
Abbildung 13: Objektanzeige 36
Abbildung 14: Einstellungen 37
Abbildung 15: Einstellung Y-Axis 37
Abbildung 16: Einstellung LineType 37
Abbildung 17: Einstellung Range 38
Abbildung 18: Einstellung Type 38
Abbildung 19: Boolean Sperre 38
Abbildung 20: Multi Seiten 39
Abbildung 21: UI Beschriftung 39
Abbildung 22: TsNet Test 40
Abbildung 23: Hilfstabelle 40
Abbildung 24: Boolean in Hilfstabelle 41
Abbildung 25: Hilfstabelle mit TestSteps und Continuous Einstellung 42
Abbildung 26: Testselektion 43
Abbildung 27: Struktogramm Trend Button 43
Abbildung 28: Steuerelemente Sperrung 44
Abbildung 29: Struktogramm Select Button 44
Abbildung 30: Struktogramm Trend Button/Diagrammerstellung Teil1 46
Abbildung 31:Struktogramm Trend Button/Diagrammerstellung Teil2 47
Abbildung 32: Beispiel Diagramm 47
Abbildung 33: Module 51

1. Teil 1

1.1 Einführung

1.1.1 IPA

Die IPA ist ein Projekt, welches in einem Zeitrahmen von 80h im Betrieb erledigt wird. Die IPA bedeutet «Individuelle Praktische Arbeit». Es wird dabei geprüft, ob ein Informatik Lehrling über das Wissen und die praktische Umsetzungsfähigkeit besitzt, ein ganzes Projekt alleine abzurollen. Die Informationen dazu erhielten wir vom Berufsbildner und den Experten und Expertinnen über das PkOrg.

1.1.2 Zweck des Dokumentes

Der IPA-Bericht ist in zwei Teile aufgeteilt. Der erste Teil besteht aus der Aufgabenstellung gemäss PkOrg, Projektorganisation, Zeitplan und Arbeitsjournal. Der zweite Teil besteht aus der Projektorganisation. Der IPA-Bericht enthält alle Tätigkeiten, die während der IPA gemacht wurden.

1.1.3 Ziel Publikum

Das Zielpublikum sind die Experten und der Fachvorgesetzte. Zu dem Zielpublikum gehört aber auch die ganze Abteilung oder andere Entwickler von TsNet, die dieses Dokument als Hilfestellung nutzen können

1.2 Projektauftrag gemäss PkOrg

Titel	Neue Funktionen Testautomationstool TsNet V2
Kandidat	Patrick Schöpfer
Fachvorgesetzter	Michael Speckien
Auftraggeber	Michael Speckien

Tabelle 1: Projektauftrag

1.2.1 Ausgangslage

TsNet wird seit geraumer Zeit zur Testspezifikation und Testdurchführung innerhalb der Siemens Gebäudetechnik genutzt.
Der Bereich Applications hat ein Excel-Sheet zur Spezifikation von Testschritten erstellt. Das Projekt TsNet V2 besteht aus
1. einem Definitionsteil (Excel-Template), in dem die Testschritte und die erwarteten Ergebnisse definiert werden sowie
2. einem Run-Time-Teil, der mit dem Controller kommuniziert und Testschritte vorgibt und Ergebnisse abfragt.

Im Rahmen von Lehrlingsarbeiten und Praktika wird der Definitionsteil komplett neu erstellt. Dazu gibt es bereits ein abgeschlossenes Excel-Projekt, das in dieser IPA mit weiteren Funktionen ergänzt werden soll.
Das Ziel dieser IPA ist es, die Testspezifikation mit Hilfe von Diagrammen darzustellen, um eine bessere Verifikation und Dokumentation zu erhalten.

1.2.2 Detaillierte Aufgabenstellung

Als Basis für die Arbeiten gilt eine Requirement-Specification (Dokumentenpool, IPA Patrick Schöpfer)
Neben der Spezifikation werden diverse Beispieldateien zur Verfügung gestellt. Diese dienen zur Erläuterung der Spezifikation und zum Test der Funktionen
Aufgabenstellung: Erstellen der Diagrammfunktion für TsNet V2 gemäss Spezifikation
- Terminplanung und Projektstatus.
- Alle 2 Tage ist ein Statusmeeting mit dem Auftraggeber durchzuführen
- Erstellung der Software
- Erstellung der Softwaredokumentation. Zielgruppe: SW-Entwickler, die das TsNet-Tool warten, pflegen und weiterentwickeln
- Erstellen der Testfälle und Durchführen von Tests.
- Test der Eingabedialoge mit korrekten und falschen Werten
- Test der Useability der Eingabebedialoge mit Fachperson ausserhalb vom TsNet-Projekt
- Test aller erstellten Funktionen gegenüber der Spezifikation
- Test aller erstellten Funktionen gegen Fehlbedienung und fehlerhafte Daten und Dateien
- Entsprechend den IPA-Regeln ist ein Arbeitsjournal zu führen.

Erwartete Lieferungen
- Terminpläne und Projektstatus alle 2 Tage und am Ende der IPA
- Excel-File mit der Erweiterung für die Diagrammfunktion
- kommentierter Source-Code
- kommentierte Eingabedialoge
- Softwaredokumentation:
Es kann jede grafische Darstellung des Programmablaufs und des Datenflusses verwendet werden,
sofern sie dazu geeignet ist, einem anderen Entwickler die Weiterarbeit zu ermöglichen.
Einfache Funktionen müssen nicht grafisch dokumentiert werden, jedoch müssen alle erstellten Funktionen
aufgelistet und detailliert beschrieben werden
- Testspezifikation und Dokumentation der Testergebnisse
- Arbeitsjournal

1.2.3 Mittel und Methoden

Entwicklungsumgebung:
- Standard-PC mit Microsoft Windows 7 oder höher
- Microsoft Excel 2007. Programmierung in VBA.
- Erstellung der Dokumentation, der Präsentation und weiterer Dokumente mit Microsoft Office 2007
- Firmen-Richtlinie Codierungsrichtlinie Visual Basic und weitere Vorgaben in der Spezifikation.

1.2.4 Vorkenntnisse

Patrick hat bereits im Vorfeld der IPA im Projekt TsNet gearbeitet, so dass ihm die Entwicklungsumgebung und das Projektumfeld bekannt sind.

1.2.5 Vorarbeiten

Mit dieser Funktion soll Patrick eine weitere Funktion zu dem bestehenden TsNet V2 hinzufügen.
Vorarbeiten für diese Funktion werden keine gemacht, Patrick fängt bei IPA-Start mit dem Umsetzen an.

1.2.6 Neue Lerninhalte

Darstellung von Diagrammen mit Excel. Komplettes Dokumentieren des Arbeitsprozesses und des Programms.

1.2.7 Arbeiten in den letzten 6 Monaten

Arbeiten an diversen kleineren Excel VBA Projekten (ca. 2 Monate) PC-Installationen (3 Wochen) Arbeiten am Projekt TsNet v2 mit Excel VBA (ca. 3 Monate).

1.3 Projektorganisation

1.3.1 Arbeitsplatz

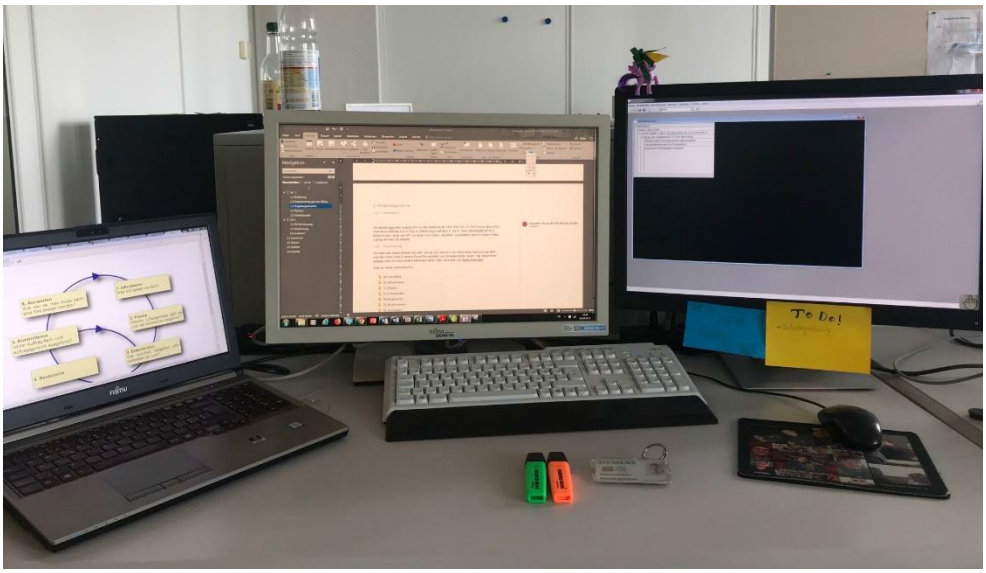


Abbildung 1: Arbeitsplatz

Ich arbeite seit dem August 2017 in der Abteilung BT CPS R&D ZG CS SAP und im Büro 425. Mein Büro befindet sich in Zug im Zählerweg 5 auf dem 4. Stock. Mein Arbeitsplatz ist mit 2 Bildschirmen, einer von HP und einer von Fujitsu, versehen. Ausserdem habe ich einen Fujitsu Laptop mit dem ich arbeite.

1.3.2 Versionierung

Alle meine Dateien wurden auf dem Server von Siemens sichergestellt. Ich habe diese Methode gewählt, weil sich mein Code in einem Excel File befindet und ich jedes Modul, jeden Tag, exportieren müsste, wenn ich eine andere Methode gewählt hätte.

Dies ist meine Ordnerstruktur:

Abbildung 2: Ordnerstruktur Server

Ich habe ausserdem alle Dateien lokal auf meinem Rechner abgespeichert, um ein eigenes Backup zu besitzen.

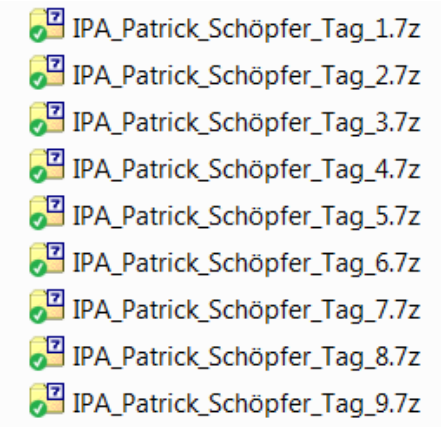


Abbildung 3: Lokale Backup Ordnerstruktur

1.3.3 Personen und Dienste

Fachvorgesetzter	Michael Speckien
Kandidat	Patrick Schöpfer
Akzeptanztester	Anton Kryenbuehl
Berufsbildung	Martin Häusler
Tester	Anton Kryenbuehl
Verwendete Software	• Microsoft Office 2016 • Snipping Tool (für Screenshots) • Struktogrammer

Tabelle 2: Personen und Dienste

1.3.4 Abteilung BT CPS &D ZG CS SAP

Gebäudeautomation bedeutet, dass sich z.B. die Heizung automatisch einstellt sobald eine gewisse Temperatur vorhanden ist. Um dies umzusetzen, brauchen wir Geräte, welche die Heizung, Storen oder Lüfter steuern. Die Abteilung «Application» erstellt die Funktionsblöcke, welche auf die Controller geladen werden. Dafür gibt es einige Tools wie z.B. ABT. Mit diesem Tool wird es möglich, diese Funktionen zu verändern oder neu zusammen zu stellen. Ein anderes Tool ist das TsNet, mit welchem man Werte simulieren kann und testen kann, ob der Controller wie erwartet reagiert.

1.3.5 Projektmanagement Methode

IPERKA

Ich habe mich für die Projektmanagementmethode IPERKA entschieden. IPERKA ist eine Methode mit 6 Schritten. Die Buchstaben von IPERKA stehen für: Informieren, Planen, Entscheiden, Realisieren, Kontrolle und Auswerten.

Wie man sieht, wird IPERKA nicht nur für die Ausführung von dem Auftrag selber verwendet, sondern enthält von der Informationsbeschaffung alles bis zur Auswertung.

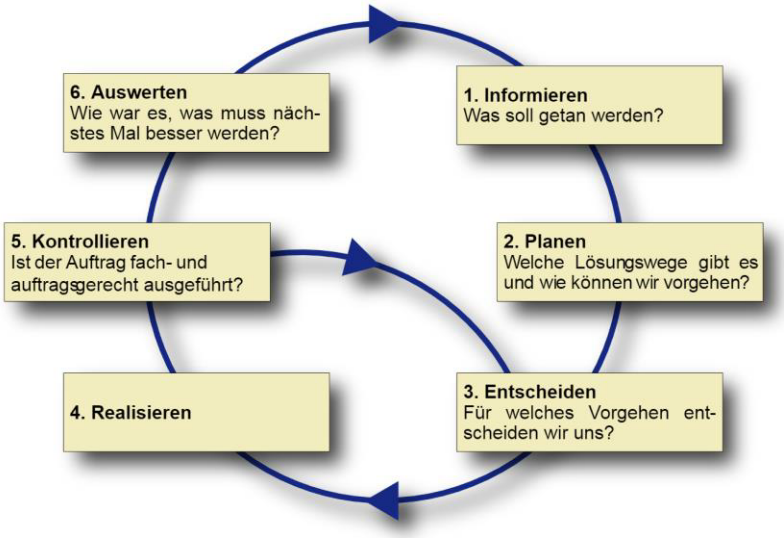


Abbildung 4: IPERKA (siehe Quellenverzeichnis)

Wie wurde IPERKA in der Arbeit angewendet

Ich habe den Zeitplan in die 6 Schritte der Methode eingeteilt. Zuerst wurden die Aufgaben den 6 Schritten zugeteilt. Durch diese Zuteilung kann ich Schritt für Schritt abarbeiten und weiss genau wann ich zum nächsten Schritt gehen kann.

Begründung der Wahl

Ich habe mich für IPERKA entschieden, weil sich IPERKA sehr gut für die selbstständige Arbeiten eignet. Man kann die Tasks im Zeitplan perfekt einordnen. Ich habe von allen Projektmanagement Methoden am meisten Erfahrung mit IPERKA, da diese schon in der Schule und in der Lernwerkstatt sehr oft angewendet wurde.

1.4 Planung

1.4.1 Soll-Zeitplan

		Aufwand	Status		Mo		Di		Mi		Do		Mo		Di		Mi		Do		Di		Mi	
Nr.	Tätigkeit	Soll (h)	Meilenstein	Status	19.03.2018		20.03.2018		21.03.2018		22.03.2018		26.03.2018		27.03.2018		28.03.2018		29.03.2018		03.04.2018		04.04.2018	
					4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h
1	Informieren																							
1.1	Dokumentation																							
1.1.1	Aufgabenstellung durchlesen und bearbeiten	0.5		i.A.	0.5																			
1.1.2	Kriterienkatalog durchlesen	0.5		i.A.	0.5																			
1.2	UI																							
1.2.1	Design regeln	0.5		i.A.	0.5																			
1.3	Programm																							
1.3.1	Verwendete Techniken Analysieren	1.0		i.A.	1.0																			
2	Planen																							
2.1	Dokumentation																							
2.1.1	Tasks und Meilensteine definieren	1.0		i.A.	1.0																			
2.1.2	Soll-Zeitplan erstellen	2.0	19.03.2018	i.A.		2.0																		
2.1.3	IPA-Bericht Inhaltsstruktur grob aufbauen	1.0		To Do			1.0																	
2.2	UI																							
2.2.1	Designumsetzung planen	2.0		To Do			2.0																	
2.3	Programm																							
2.3.1	Diagramm-Funktion analysieren	1.0		To Do			1.0																	
2.3.2	Planungsdiagramme erstellen (Struktogramm, Flussd., etc)	4.0		To Do				2.0	2.0															
2.3.3	Testkriterien definieren	2.0		To Do					2.0															
3	Entscheiden																							
3.1	Dokumentation																							
3.1.1	Struktur und Gliederung festlegen	0.5		To Do						0.5														
3.2	UI																							
3.2.1	Für UI-Umsetzungsmethode entscheiden	0.5		To Do						0.5														
4	Realisieren																							
4.1	UI																							
4.1.1	Design aufbauen	2.0		To Do							2.0													
4.1.2	Objektanzahl dynamisch anpassbar	2.0		To Do							2.0													
4.1.3	UI inkl. Einstellungen speicherbar	1.0		To Do								1.0												
4.2	Programm																							
4.2.1	Hilfstabelle implementieren	4.5		To Do								1.5	3.0											
4.2.2	Diagramm erstellung implementieren	2.0		To Do									1.0	1.0										
4.3	Dokumentation																							
4.3.1	Software Dokumentation	3.0		To Do											3.0									
4.3.2	Summary	2.0	27.03.2018	To Do											1.0	1.0								
4.3.3	Testfälle schreiben/anpassen	1.5		To Do												0.5	1.0							
5	Kontrollieren																							
5.1	Dokumentation																							
5.1.1	Software Dokumentation kontrollieren	1.0		To Do													1.0							
5.2	Programm																							
5.2.1	Tests durchführen	3.5	28.03.2018	To Do													1.0	1.5	1.0					
5.2.2	Fehlerbehebung Tests	2.0		To Do															2.0					
5.2.3	Nachtests durchführen	2.0		To Do															1.0	1.0				
6	Auswerten																							
6.1	Programm																							
6.1.1	Tests für Schlussbericht auswerten	1.5		To Do																0.5	1.0			
6.2	Dokumentation																							
6.2.1	Schlussbericht verfassen	1.0		To Do																	1.0			
6.2.2	Rework Flussdiagramm	1.0		To Do																	1.0			
7	Diverses																							
7.1	Gespräch Akzeptanztester	1.5		To Do	0.5												1.0							
7.2	Statusmeeting mit Fachvorgesetzter	2.5		To Do		0.5				0.5				0.5				0.5				0.5		
7.3	Ist-Zeitplan führen	5.0		To Do		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5
7.4	Arbeitsjournal	5.0		To Do		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5
7.5	IPA Bericht	13.0		To Do				0.5		1.0				1.0		1.0		0.5		1.0	1.0	2.0	4.0	1.0
7.6	IPA drucken, binden, etc.	1.5	04.04.2018	To Do																				1.5
8	Reseve																							
8.1	Pufferzeit (10 * 0.5)	5.0				0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5
	Total	**80.0**			**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**

Abbildung 5: Soll Zeitplan

1.4.2 Soll-Ist-Vergleich

		Aufwand		Status		Mo		Di		Mi		Do		Mo		Di		Mi		Do		Di		Mi	
Nr.	Tätigkeit	Soll (h)	Ist (h)	Meilenstein	Status	19.03.2018		20.03.2018		21.03.2018		22.03.2018		26.03.2018		27.03.2018		28.03.2018		29.03.2018		03.04.2018		04.04.2018	
						4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h	4h
1	Informieren																								
1.1	Dokumentation																								
1.1.1	Aufgabenstellung durchlesen und bearbeiten	0.5	0.5		Done	0.5																			
1.1.2	Kriterienkatalog durchlesen	0.5	0.5		Done	0.5																			
1.2	UI																								
1.2.1	Design regeln	0.5	0.5		Done	0.5																			
1.3	Programm																								
1.3.1	Verwendete Techniken Analysieren	1.0	1.0		Done	1.0																			
2	Planen																								
2.1	Dokumentation																								
2.1.1	Tasks und Meilensteine definieren	1.0	1.0		Done	1.0																			
2.1.2	Soll-Zeitplan erstellen	2.0	2.0	19.03.2018	Done		2.0																		
2.1.3	IPA-Bericht Inhaltsstruktur grob aufbauen	1.0	1.0		Done			1.0																	
2.2	UI																								
2.2.1	Designumsetzung planen	2.0	2.0		Done			2.0																	
2.3	Programm																								
2.3.1	Diagramm-Funktion analysieren	1.0	1.0		Done			1.0																	
2.3.2	Planungsdiagramme erstellen (Struktogramm, Flussd., e	4.0	4.0		Done				2.0	2.0															
2.3.3	Testkriterien definieren	2.0	2.0		Done					2.0															
3	Entscheiden																								
3.1	Dokumentation																								
3.1.1	Struktur und Gliederung festlegen	0.5	0.5		Done						0.5														
3.2	UI																								
3.2.1	Für UI-Umsetzungsmethode entscheiden	0.5	0.5		Done						0.5														
4	Realisieren																								
4.1	UI																								
4.1.1	Design aufbauen	2.0	2.0		Done							2.0													
4.1.2	Objektanzahl dynamisch anpassbar	2.0	2.0		Done							2.0													
4.1.3	UI inkl. Einstellungen speicherbar	1.0	1.0		Done								1.0												
4.2	Programm																								
4.2.1	Hilfstabelle implementieren	4.5	4.5		Done								1.5	3.0											
4.2.2	Diagramm erstellung implementieren	2.0	3.0		Done									1.0	1.0	1.0									
4.3	Dokumentation																								
4.3.1	Software Dokumentation	3.0	3.0		Done											3.0									
4.3.2	Summary	2.0	1.5	27.03.2018	Done												1.5								
4.3.3	Testfälle schreiben/anpassen	1.5	1.5		Done												0.5	1.0							
5	Kontrollieren																								
5.1	Dokumentation																								
5.1.1	Software Dokumentation kontrollieren	1.0	1.0		Done													1.0							
5.2	Programm																								
5.2.1	Tests durchführen	3.5	3.5	28.03.2018	Done													1.0	1.5	1.0					
5.2.2	Fehlerbehebung Tests	2.0	2.0		Done															2.0					
5.2.3	Nachtests durchführen	2.0	2.0		Done															1.0	1.0				
6	Auswerten																								
6.1	Programm																								
6.1.1	Tests für Schlussbericht auswerten	1.5	1.5		Done																0.5	1.0			
6.2	Dokumentation																								
6.2.1	Schlussbericht verfassen	1.0	1.0		Done																	1.0			
6.2.2	Rework Flussdiagramm	1.0	1.0		Done																	1.0			
7	Diverses																								
7.1	Gespräch Akzeptanztester	1.5	1.5		Done	0.5												1.0							
7.2	Statusmeeting mit Fachvorgesetzter	2.5	2.5		Done		0.5				0.5				0.5				0.5				0.5		
7.3	Ist-Zeitplan führen	5.0	5.0		Done		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5
7.4	Arbeitsjournal	5.0	5.0		Done		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5		0.5
7.5	IPA Bericht	13.0	13.0		Done				0.5		1.0				1.0		1.0		0.5		1.0	1.0	2.0	4.0	1.0
7.6	IPA drucken, binden, etc.	1.5	1.5	04.04.2018	i.A.																				1.5
8	Reseve																								
8.1	Pufferzeit (10 * 0.5)	5.0	4.5				0.0		0.0		0.0		0.5		1.0		1.0		1.0		0.5		0.5		0.0
	Total	**80.0**	80.0			**4.0**	**3.5**	**4.0**	**3.5**	**4.0**	**3.5**	**4.0**	**4.0**	**4.0**	**4.5**	**4.0**	**5.0**	**4.0**	**4.5**	**4.0**	**4.0**	**4.0**	**4.0**	**4.0**	**3.5**

Abbildung 6: Soll-Ist-Zeitplan

1.4.3 Tätigkeiten

Nr.	Tätigkeit	Beschreibung
1.1.1	Aufgabenstellung durchlesen und bearbeiten	Die detaillierte Aufgabenstellung online auf PkOrg genau durchlesen und verstehen. Ich werde hier handschriftliche Notizen erstellen um mir einen Überblick über die Aufgaben zu schaffen.
1.1.2	Kriterienkatalog durchlesen	Ich werde den Kriterienkatalog durchlesen um einen Überblick zu bekommen, was wie viele Punkte für die Bewertung gibt. Ich werde diese am Ende des Projektes nochmals genau durchlesen.
1.2.1	Design regeln	Ich werde die Designregeln nochmals nachschauen, da ich ein UI erstellen muss und die Designvorschriften einhalten möchte.
1.3.1	Verwendete Techniken analysieren	Ich werde die Techniken die ich vergangenes Jahr gelernt habe und notiert habe nochmals kurz durchgehen und wichtiges markieren.
2.1.1	Tasks und Meilensteine definieren	Ich werde aufgrund von den gesammelten Informationen Tasks und Meilensteine definieren. Dazu werde ich mir genau überlegen für welche Tätigkeiten ich wie viel Zeit benötige.
2.1.2	Soll-Zeitplan erstellen	Ich werde die Tasks und Meilensteine in einen Zeitplan umwandeln. Ich werde eine absolute Zeitachse hinzufügen und sie in 4h Blöcke einteilen.
2.1.3	IPA-Bericht Inhaltsstruktur grob aufbauen	Ich werde planen, was alles in den IPA-Bericht gehört und erstelle Titel und Überschriften
2.2.1	Designumsetzung planen	Ich plane wie ich das Design umsetzen möchte und mache mir Notizen und Prototypen.
2.3.1	Diagramm-Funktion analysieren	Ich werde planen, wie ich das Diagramm erstellen werde. Dazu schaue ich mir Beispielcodes im Internet an.
2.3.2	Planungsdiagramme erstellen (Struktogramm, Flussd., etc)	Ich werde Diagramme erstellen, die den Ablauf des Programms zeigen. Das Ziel der Diagramme ist, dass die Implementation schneller und einfacher verläuft.
2.3.3	Testkriterien definieren	Ich definiere die Testkriterien für die späteren Testfälle.
3.1.1	Struktur und Gliederung festlegen	Ich gehe in der Dokumentation nochmals die Gliederung und die Struktur durch, die ich geplant habe und entscheide mich für die endgültige Fassung.
3.2.1	Für UI-Umsetzungsmethode entscheiden	Ich werde mich für eine geeignete Umsetzungsmethode entscheiden. Ich

		werde hier den Fachvorgesetzten miteinbeziehen, da das Programm ja weiterverwendet wird.
4.1.1	Design aufbauen	Ich werde zuerst das User Interface implementieren.
4.1.2	Objektanzahl dynamisch anpassbar	Die Objektanzahl, die ausgewählt wird, wird dynamisch ins UI gefüllt
4.1.3	UI inkl. Einstellungen speicherbar	Ich werde implementieren, dass man die gewählten Einstellungen abspeichern kann, damit man das Diagramm immer wieder erstellen kann.
4.2.1	Hilfstabelle implementieren	Ich werde die Hilfstabelle implementieren. Die Hilfstabelle beinhaltet alle Daten die zum Diagramm beitragen.
4.2.2	Diagrammerstellung implementieren	Die Diagrammerstellung beschreibt den Aufbau des automatisch generierten Diagrammes
4.3.1	Softwaredokumentation	Ich werde eine Softwaredokumentation schreiben. Sie ist für andere Entwickler oder z.B. neue Lehrlinge in der Abteilung gedacht, welche möglicherweise mit dieser Funktion arbeiten oder sie erweitern werden.
4.3.2	Summary	Das Summary ist eine etwa einseitige Zusammenfassung.
4.3.3	Testfälle schreiben/anpassen	Ich schreibe Testfälle und passe sie an, um mein Programm später testen zu können.
5.1.1	Softwaredokumentation kontrollieren	Ich lese meine Softwaredokumentation nochmals durch, überprüfe Rechtschreibfehler etc.
5.2.1	Tests durchführen	Ich führe meine Tests durch. Ich werde nach den Tests etwa sehen können, bei welchem Stand ich bin.
5.2.2	Fehlerbehebung Tests	Ich werde mir noch einmal Zeit nehmen, die fehlgeschlagenen Tests zu verbessern.
5.2.3	Nachtests durchführen	Ich werde die fehlgeschlagenen Tests nochmals durchgehen um zu schauen ob nun alles korrekt ist.
6.1.1	Tests für Schlussbericht auswerten	Selbsterklärend
6.2.1	Schlussbericht verfassen	Selbsterklärend
6.2.2	Rework Flussdiagramm	Ich werde die Diagramme nochmals anpassen, dass auch der nächste Entwickler ein sauberes Struktogramm und Flussdiagramm hat
7.1	Gespräch Akzeptanztester	Das sind Gespräche/Besprechungen, welche ich mit dem Akzeptanztester halte.
7.2	Statusmeeting mit Fachvorgesetzter	Ich werde mit dem Fachvorgesetzten jeden 2. Tag ein Statusmeeting halten, damit auch er über meinen Stand Bescheid weiss.
7.3	Ist-Zeitplan führen	Der Ist-Zeitplan ist jeden Tag sauber nach zu führen.

7.4	Arbeitsjournal	Das Arbeitsjournal beschreibt meine Tätigkeiten, Schwierigkeiten und mögliche Verschiebungen im Zeitplan.
7.5	IPA-Bericht	Diese Dokumentation ist der IPA-Bericht
7.6	IPA drucken, binden, etc.	Selbsterklärend
8.1	Pufferzeit (10 * 0.5)	Meine Reserve Zeit

Tabelle 3: Tätigkeiten Beschreibung

1.4.4 Meilensteine

Ich habe für das ganze Projekt 4 Meilensteine eingeplant. Die Meilensteine wurden so eingeplant, dass alles bis zum Enddatum fertig ist.

Meilenstein	**Begründung für Meilenstein / Erklärung**	**Arbeitsschritte**
19.03.2018	Ich habe diesen Meilenstein gewählt, da der Zeitplan sowieso am 1. Tag fällig ist und ich finde, dass der Zeitplan ein Grundstein für das ganze Projekt ist.	1.1.1 bis 2.1.2
27.03.2018	Dies ist der 2. Meilenstein. Er steht für alle praktische Arbeiten ausser den Testfällen.	2.1.3 bis 4.3.2
28.03.2018	Der 3. Meilenstein steht für die Tests. Alles muss bis zu den Tests erledigt sein, da man sonst nicht richtig testen kann.	4.3.3 bis 5.2.1
04.04.2018	Der 4. Meilenstein sind die letzten Arbeiten und die Abgabe der IPA	5.2.2 bis 7.6 ausgenommen 7.1 bis 7.4. Diese sind nicht Meilenstein relevant, da sie meist täglich oder alle zwei Tage statt fanden

Tabelle 4: Meilensteine

1.4.5 Visualisierung der Arbeitsschritte

Ich habe für die Visualisierung der Arbeitsschritte Flussdiagramme und Struktogramme erstellt. Die Flussdiagramme habe ich für einen ersten Überblick hier eingefügt und die Struktogramme im Kapitel «Realisieren» zu den zugehörigen Funktionen. Das 1. Flussdiagramm erklärt den Aufruf aus dem Overview. Ab dem «Trend» Button im Overview wurde alles von mir erstellt. Das 2. Flussdiagramm erklärt den ganzen Ablauf meines Programmes.

Flussdiagramm Trend Button im Overview

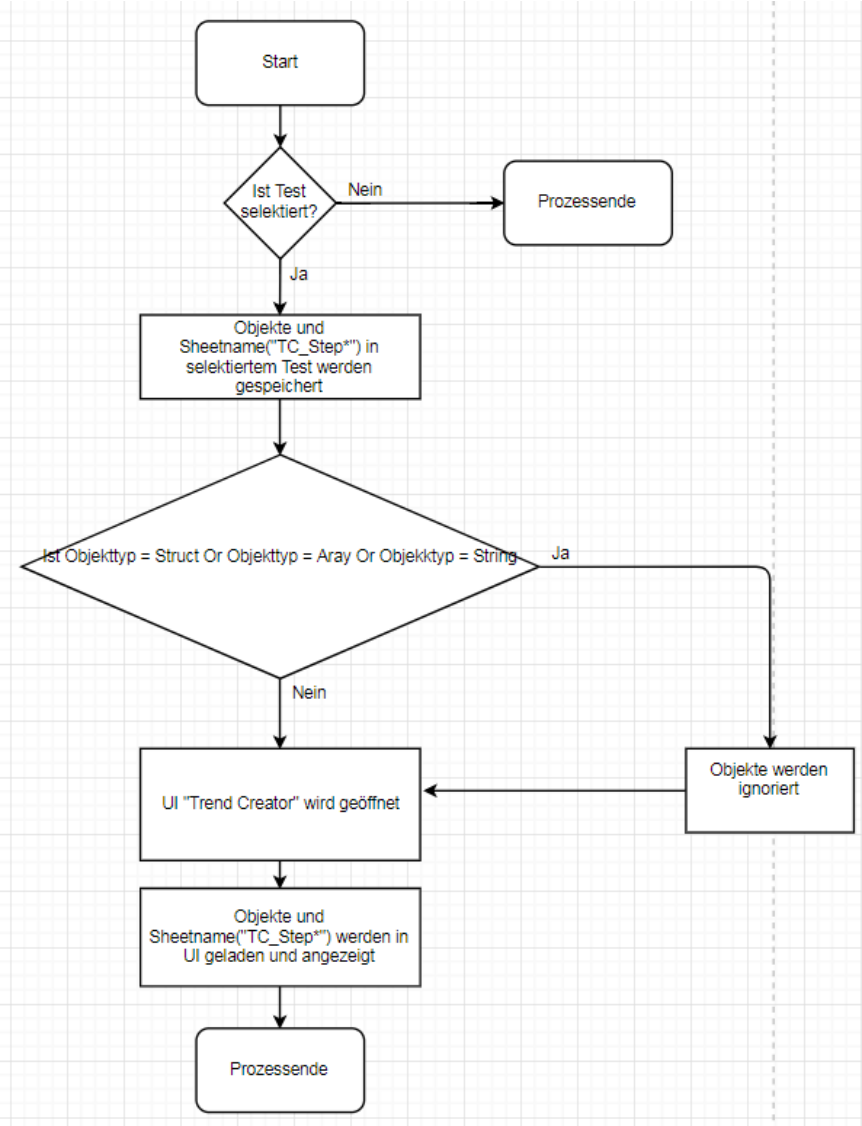


Abbildung 7: Flussdiagramm Trend Button

Flussdiagramm Trend Create Button im UI

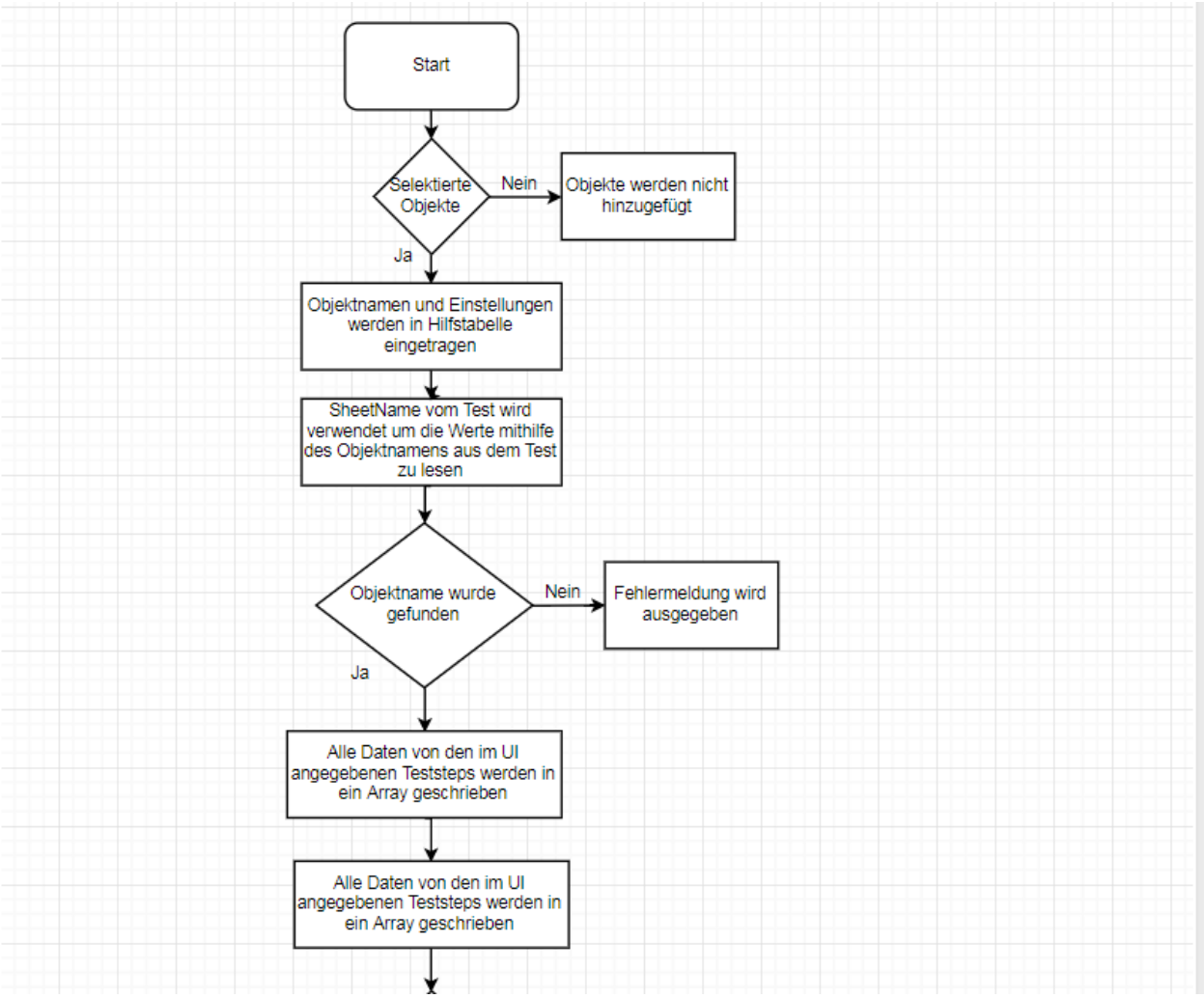


Abbildung 8: Flussdiagramm Create Trend Teil1

Fortsetzung folgt auf der nächsten Seite.

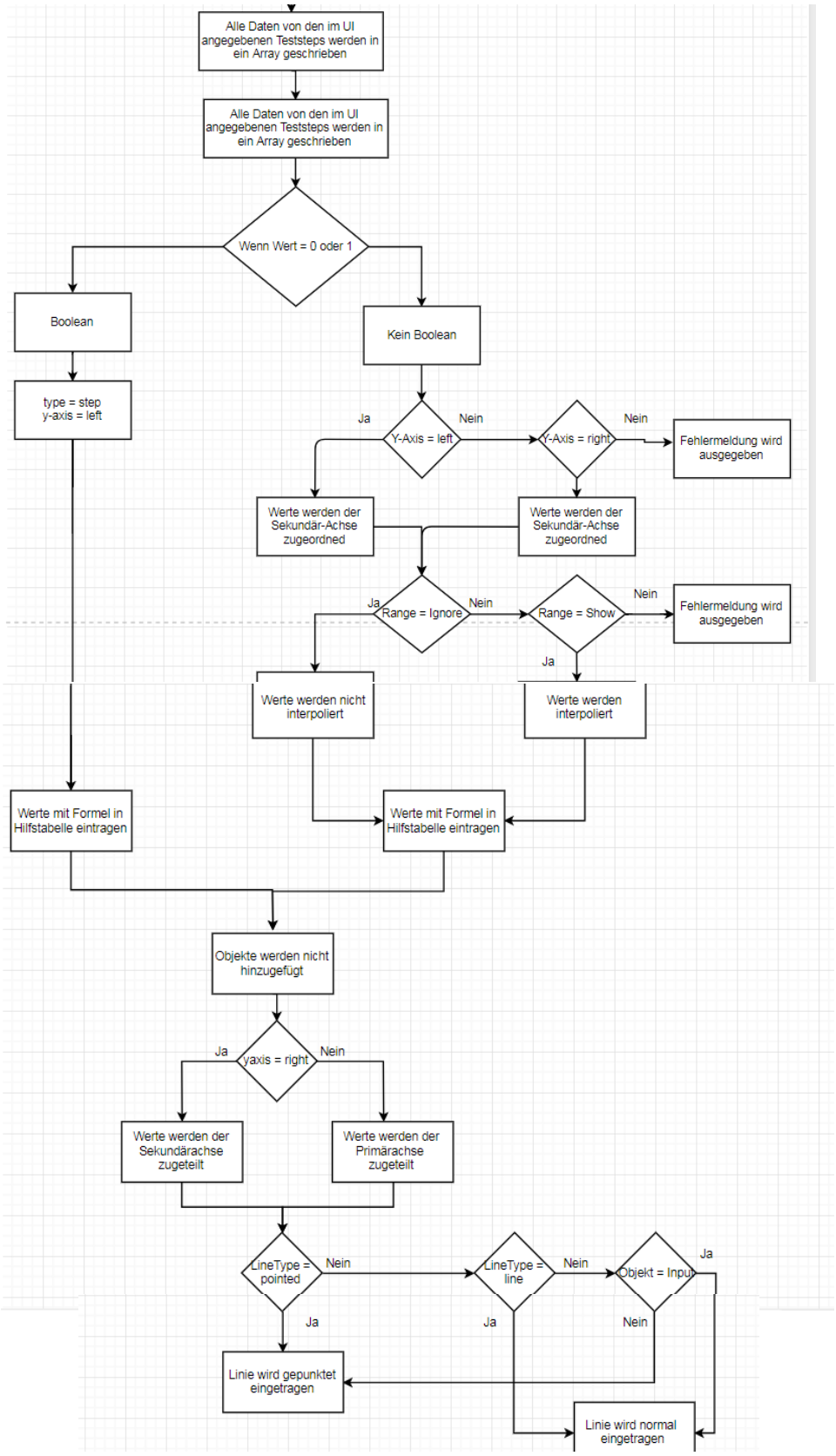


Abbildung 9: Flussdiagramm Create Trend Teil2

1.4.6 Erklärung Flussdiagramm

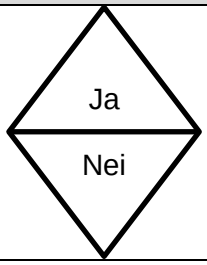
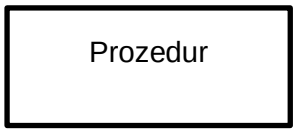
Begriff/Form	Erklärung
	Eine Wenn oder If Verzweigung
	Prozess Start, Prozess Ende
	Einzelne Prozedurschritte

Tabelle 5: Flussdiagramm Erklärung

1.5 Arbeitsjournal

1.5.1 Zweck

Im Arbeitsjournal befinden sich alle Tätigkeiten. Es sind Probleme und Hilfestellungen, sowie Besprechungen enthalten. Die Journale sind der Nr. des Zeitplans zugeordnet.

1.5.2 Anwendungsbereich

Das Arbeitsjournal ist keine Dokumentation und ersetzt auch keine Dokumentation. Die Arbeitsjournale beinhalten Tätigkeiten, allfällige Hilfestellungen, Schwierigkeiten, Verzögerungen, Besprechungen, die nächsten Schritte.

1.5.3 Aufbau/Legende

Das Arbeitsjournal ist folgendermassen aufgebaut:

Allgemeine Informationen:

Element	Beschreibung
Datum	Das Datum, für welches das Arbeitsjournal geschrieben wurde.
Status	An welchem Ort ich mich im Zeitplan befinde.
Hilfestellung	Welche Hilfestellungen von welchen Personen ich bekommen habe.
Mailverkehr	Die Personen mit denen ich an diesem Tag geschrieben habe.
Besprechungen	Mit welchen Personen ich Besprechungen hatte.

Tabelle 6: Arbeitsjournal allgemeine Informationen

Information zu den Tätigkeiten:

Element	Beschreibung
ID	Die Tätigkeitsnummer referenzierend zum Zeitplan.
Tätigkeit	An welchem Ort ich mich im Zeitplan befinde.
Soll(h)	Die Zeit in Stunden, welche insgesamt für diese Tätigkeit an diesem Tag eingeplant wurde.
Ist(h)	Die Zeit in Stunden, welche insgesamt für die Tätigkeit an diesem Tag aufgewendet wurde.

Tabelle 7: Arbeitsjournal Informationen zu den Tätigkeiten

Besprechungen:

In diesem Abschnitt wird festgehalten, was in den Besprechungen erwähnt wurde.

Next Steps:

In diesem Abschnitt wird festgehalten, was die nächsten Schritte sind.

1.5.4 Arbeitsjournale

Datum		19.03.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Anton Kryenbuehl		
Besprechungen		Anton Kryenbuehl, Michael Speckien		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
1.1.1	Aufgabenstellung durchlesen und bearbeiten.	Abgeschlossen	0.5	0.5
1.1.2	Kriterienkatalog durchlesen	Abgeschlossen	0.5	0.5
1.2.1	Design Regeln durchgehen	Abgeschlossen	0.5	0.5
1.3.1	Verwendete Techniken analysieren	Abgeschlossen	1.0	1.0
2.1.1	Tasks und Meilensteine definieren	Abgeschlossen	1.0	1.0
2.1.2	Soll-Zeitplan erstellen	Abgeschlossen	2.0	2.0
7.3	Ist-Zeitplan	In Arbeit	0.5	0.5
7.4	Arbeitsjournal	In Arbeit	0.5	0.5
Besprechung		**Personen**		
Statusmeeting		Michael Speckien		
Tester Meeting		Anton Kryenbuehl		

Tätigkeiten:

Heute habe ich mit meiner IPA gestartet. Ich habe mich zuerst eingerichtet und bin gleich auf PkOrg um die Aufgabenstellung genauer durchzulesen. Ich war mir nicht ganz sicher, was bei Useability Tests wichtig ist, deshalb habe ich im Internet dazu recherchiert und mir Notizen zu den Ergebnissen gemacht. Als ich dies abgeschlossen hatte, habe ich den Kriterienkatalog durchgelesen und mir nebenbei handschriftlich Notizen gemacht. Ich habe mir danach überlegt, welche Tasks ich habe. Mir ist schnell aufgefallen, dass ich ein UI erstellen muss. Ich konnte mich erinnern, dass es bei der Erstellung von UIs wichtig ist, auf bestimmte Regeln zu achten, wie zum Beispiel angemessene Grössen von Eingabefelder etc. Ich habe analysiert, was ich bis jetzt für Techniken in VBA kenne, welche nützlich für die künftigen Aufträge sein könnten und habe mich darüber informiert. Ich habe die definierten Tasks bestimmten Meilensteinen zugeordnet und daraus den Zeitplan erstellt. Die Meilensteine sind so gewählt, dass ich einen Abgabetermin einhalten muss, oder sonst wichtig für den Zeitplan ist. Da ich mit IPERKA arbeite habe ich den Zeitplan in die bekannten 6 Schritte unterteilt. Als ich damit fertig war habe ich die Tasks noch dem Zuständigkeitsort hinzugefügt, das heisst, dass ich innerhalb von den Schritten noch zwischen UI, Programm und Dokumentation unterschieden habe. Am Ende des Tages habe ich den Zeitplan meiner Erstexpertin per E-Mail geschickt und ihn auf PkOrg als «Hinweis an die Experten» gesendet.

Besprechungen:

Ich hatte heute zwei Besprechungen. Die 1. Besprechung hatte ich mit Anton Kryenbuehl. In der Aufgabenstellung steht, dass ich eine 3. Person brauche, die die Useability meines Programmes testet. Ich habe mit ihm darüber gesprochen und er erklärte sich einverstanden dies für mich zu machen. Die 2. Besprechung hatte ich mit meinem Fachvorgesetzten, Michael Speckien, ich habe ihm meinen Zeitplan und meinen aktuellen Stand präsentiert. Danach arbeitete ich weiter am Projekt.

Next Steps:

Morgen werde ich das ganze Projekt Hauptsächlich planen und mich genauer in die Aufgabenstellung reinversetzen.

Arbeitsjournal Tag 2

Datum		20.03.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Keine		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
2.1.3	IPA-Bericht Inhaltsstruktur grob aufbauen	Abgeschlossen	1.0	1.0
2.2.1	Designumsetzung planen	Abgeschlossen	2.0	2.0
2.3.1	Diagramm Funktion Analysieren	Abgeschlossen	1.0	1.0
2.3.2	Planungsdiagramme erstellen (Struktogramm, Flussdiagramm)	In Arbeit	2.0	2.0
7.3	Ist-Zeitplan führen	In Arbeit	0.5	0.5
7.4	Arbeitsjournal	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	0.5	0.5
2.1.3	IPA-Bericht Inhaltsstruktur grob aufbauen	Abgeschlossen	1.0	1.0
Besprechung		**Personen**		
keine		niemand		

Tätigkeiten:

Ich habe heute weiter an meiner IPA gearbeitet. Durch meinen Zeitplan konnte ich ohne Probleme weiterarbeiten und wusste sofort, wo ich mich gerade befand. Am Morgen damit begann ich meinen IPA-Bericht grob zu strukturieren. Das bedeutet, ich habe mir handschriftliche Notizen gemacht, wie ich meinen Bericht aufbauen möchte und habe die Überschriften und Titel in ein Word-Dokument geschrieben. Das Ganze bereitete mir keine Schwierigkeiten, da ich schon sehr viel Erfahrung mit den Office Programmen sammeln konnte. Ich ging weiter zur Designumsetzung. Ich habe mir verschiedene Wege überlegt, wie ich das UI umsetzen könnte. Ein Weg war, ein User Form in VBA zu erstellen, jedoch war die Umsetzung nicht einfach, da ich wusste, dass ich in diesem User Form später eine dynamische Anzahl an Objekten darstellen muss und dies nicht übersichtlich aussehen könnte. Ich entschied mich dann für den Weg über ein Excel Tabellenblatt. Der Grund, warum ich mich dafür entschieden habe ist, da sich dieser Teil des TsNets schon in einer Excel Datei befindet und man viele Manipulationen an Tests vornimmt, würde ein Tabellenblatt für die Erstellung von Diagrammen durchaus Sinn ergeben.
Als nächstes habe ich mich im Internet zu der Diagramm Funktion eingelesen. Es fiel mir Anfangs schwer, zu verstehen welcher Befehl was bewirkt, also habe ich die Makro Aufnahmefunktion von Excel gestartet, ein einfaches Diagramm erstellt und mir den Code angeschaut, welcher generiert wurde. Man kann diesen Code nicht weiterverwenden, da er sehr statisch war, jedoch half es mir die Befehle die ich im Internet gefunden habe zu verstehen.
Danach habe ich Flussdiagramme und Struktogramme erstellt. Ich hatte am Anfang Schwierigkeiten, die Struktogramme zu erstellen also habe ich mit den Flussdiagrammen angefangen, welche mir bei den Struktogrammen sehr geholfen haben.

Next Steps:

Morgen werde ich weitermachen mit den Struktogrammen und den Flussdiagrammen. Ausserdem stehen morgen die Testkriterien Definition auf dem Plan. Am Nachmittag werde ich mich noch definitiv für die UI Umsetzungsmethode entscheiden und die Struktur und die Gliederung der Dokumentation anpassen und festlegen.

Tag 3

Datum		21.03.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Michael Speckien		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
2.3.2	Planungsdiagramme erstellen	Abgeschlossen	2.0	2.0
2.3.3	Testkriterien definieren	Abgeschlossen	2.0	2.0
3.1.1	Struktur und Gliederung festlegen	Abgeschlossen	0.5	0.5
3.2.1	Für UI-Umsetzungsmethode entscheiden	In Arbeit	0.5	0.5
7.3	Ist-Zeitplan	In Arbeit	0.5	0.5
7.4	Arbeitsjournal	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	1.0	1.0
2.3.2	Planungsdiagramme erstellen	Abgeschlossen	2.0	2.0
Besprechung		**Personen**		
Statusmeeting		Michael Speckien		

Tätigkeiten:

Als ich heute Morgen zur Arbeit gekommen bin, habe ich sofort mit Planungsdiagrammen weiter gemacht. Ich musste noch etwas Kleines an den Struktogrammen abändern, dass mir aufgefallen ist und habe dann die Struktogramme weiter bearbeitet. Ich habe sie nun fertiggestellt und bin zufrieden damit. Ich habe weiter gemacht, indem ich die Kriterien und Aufgabenstellung erneut durchgelesen habe und mir nochmal alle Kriterien für das Programm durchgelesen, mir die Planungsdiagramme angeschaut und mir Notizen dazu gemacht. Daraus habe ich dann angefangen einzelne Tests zu schreiben und diese den Erwartungen anzupassen. Ich habe die Tests in 3 Kapitel eingeteilt. Die 3 Kapitel unterscheiden sich dabei, dass im 1. Kapitel hauptsächlich Eingabedialoge geprüft werden, im 2. Kapitel hauptsächlich Funktionalität und im 3. Kapitel, in welchem sich keine Tests, sondern Fragen befinden, die Useability des Programms getestet wird.

Am Nachmittag habe ich mich zuerst für die Struktur und Gliederung entschieden. Dafür habe ich meine Notizen dazu durchgelesen und mich festgelegt. Ich hatte daraufhin ein Statusmeeting mit Michael Speckien und habe abgeklärt, ob das User Interface so wie ich es gestaltet habe in Ordnung für ihn ist (siehe Besprechungen). Ich habe ausserdem an dem IPA-Bericht gearbeitet und einen ersten Inhalt eingefügt. Ich bin momentan gut im Zeitplan.

Besprechungen:

Ich habe mit Michael über das UI gesprochen. Wir haben uns gemeinsam Gedanken darüber gemacht und ich habe vorgeschlagen, dass nicht mehr als 10 Objekte maximal in einem Diagramm dargestellt werden soll, da es sonst unübersichtlich sein könnte. Wir haben uns darauf geeinigt und ich habe mich dann für ein User Form statt Excel Tabellenblatt entschieden, da 10 Objekte problemlos darin dargestellt werden können. Damit waren beide einverstanden.

Next Steps:

Ich werde morgen einen Prototypen des User Interfaces aufstellen und mit der Implementation beginnen.

Tag 4

Datum		22.03.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Keine		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
4.1.1	Design Aufbauen	Abgeschlossen	2.0	2.0
4.1.2	Objektanzahl dynamisch	Abgeschlossen	2.0	2.0
4.1.3	UI inkl. Einstellungen speicherbar	Abgeschlossen	1.0	1.0
4.2.1	Hilfstabelle implementieren	In Arbeit	1.5	1.5
7.3	Ist-Zeitplan führen	In Arbeit	0.5	0.5
7.4	Arbeitsjournal schreiben	In Arbeit	0.5	0.5
Besprechung		**Personen**		
Keine		Niemand		

Tätigkeiten:

Ich habe heute Morgen angefangen einen Prototyp des Designs zu erstellen. Ich habe es angefangen aufzubauen und dabei bedacht, dass 4 Diagramme erstellt werden müssen. Ich habe das Steuerelement «Multi Site» verwendet um zwischen den einzelnen Diagrammen hin und her schalten zu können. Ich habe ausserdem alles andere Steuerelemente eingefügt und habe mir überlegt, wie ich die Auswahl der 10 Objekte hinzugefügt. Ich habe eine «ListBox» verwendet, welche die ganze Objekte auflistet die sich in dem selektierten Test befindet. Ich habe zusätzlich eine Spalte gemacht für die «CheckBox» Elemente. Somit habe ich die dynamische Objektanzahl sichergestellt. Da ich aus den Einstellungen eine Hilfstabelle generiere, habe ich beschlossen, die Hilfstabelle mit einem Namen zu versehen und automatisch abzuspeichern. Ich hatte einige Probleme mit der Benennung der einzelnen Steuerelemente und habe diese Zeit auch nicht eingeplant, deshalb rechne ich diese Zeit heute zu meinem Buffer.
Ich habe zuerst den Code für das Aufrufen des UIs geschrieben. Weiterhin habe ich implementiert, dass die «ListBox» mit den Objekten des Tests aufgefüllt wird. Ich habe einen Button erstellt, der beim daraufklicken eine Funktion ausführt, welche die ausgewählten Objektnamen links neben die «ComboBox» - Steuerelemente einfügt. Die Funktion füllt diese Combobox Steuerelemente mit den Einstellungsmöglichkeiten auf. Ich habe dann angefangen die Hilfstabelle zu implementieren. Ich bin nicht ganz fertig damit geworden aber laut meinem Zeitplan muss ich das auch erst am Montag sein.

Next Steps:

Am Montag werde ich die Hilfstabelle Fertig implementieren. Ausserdem werde ich am Montag die Diagramm Implementierung vornehmen und habe am Nachmittag ein Statusgespräch mit Michael.

Tag 5

Datum		26.03.2018		
Status		Nach Zeitplan		
Hilfestellung		Michael Speckien		
Mailverkehr		Kein		
Besprechungen		Michael Speckien		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
4.2.1	Hilfstabelle Implementieren	Abgeschlossen	3.0	3.0
4.2.2	Diagrammerstellung implementieren	In Arbeit	2.0	2.0
7.3	Ist-Zeitplan erstellen	In Arbeit	0.5	0.5
7.4	Arbeitsjournal erstellen	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	1.0	1.0
Besprechung		**Personen**		
Statusmeeting		Michael Speckien		

Tätigkeiten:

Ich habe heute weiter implementiert. Ich konnte die Hilfstabelle korrekt fertig implementieren, hatte aber einige Schwierigkeiten mit den Boolean Werten und dem Interpolieren. Ich habe mich mit Michael Speckien zusammengesetzt und bin mit ihm die Formeln durchgegangen, mit denen man die Werte berechnet. Diese Zeit habe ich zum Buffer gezählt. Als wir zu einer Lösung gekommen sind habe ich weiter gemacht. Durch die Schwierigkeiten ist es trotz Buffer zu kleineren Verzögerungen in meinem Zeitplan gekommen. Ich habe mit der Diagrammerstellung begonnen, konnte diese jedoch nicht ganz abschliessen. Ich werde mir morgen nochmals eine Stunde Zeit dafür nehmen und schaue, dass ich die Verzögerung abarbeiten kann.
Ich habe ausserdem den IPA-Bericht ein wenig erweitert. Ich habe Informationen zu meinem Arbeitsplatz und Organisation angegeben. Ich hatte bisher keine Probleme mit der Dokumentation.

Besprechungen:

Ich habe heute ein Statusmeeting mit Michael und habe ihm meinen Stand im Programm und Dokumentation gezeigt. Ich habe ihn in Kenntnis gesetzt, dass ich im Zeitplan ein bisschen verspätet bin.

Next Steps:

Ich werde die Implementierung abschliessen und mit der Softwaredokumentation anfangen. Danach werde ich mit meinem Summary beginnen um meine Meilensteine einzuhalten.

Tag 6

Datum	27.03.2018

Status		Nach Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Keine		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
4.2.2	Diagrammerstellung implementieren	Abgeschlossen	1.0	1.0
4.3.1	Softwaredokumentation	Abgeschlossen	3.0	3.0
4.3.2	Summary	Abgeschlossen	2.0	1.5
4.3.3	Testfälle schreiben/anpassen	In Arbeit	0.5	0.5
7.3	Ist-Zeitplan	In Arbeit	0.5	0.5
7.4	Arbeitsjournal	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	1.0	1.0
Besprechung		**Personen**		
Keine		Niemand		

Tätigkeiten:

Ich habe die Implementierung mit einer kleinen Verspätung beendet und konnte mit der Softwaredokumentation anfangen. Ich hatte am Anfang etwas Mühe, weil ich nicht genau wusste was alles in eine Softwaredokumentation gehört. Nach erneutem lesen der Aufgabenstellung bezüglich Softwaredokumentation habe ich gesehen, dass die Softwaredokumentation für Entwickler ist, das bedeutet, dass ich sie so schreiben muss, dass ein Entwickler möglichst schnell in meinen Code kommt falls er diesen erweitern muss. Ich habe also alle Funktionen erklärt und wo sie sich befinden (siehe Kapitel «Realisierung»).
Nach der Softwaredokumentation habe ich mit dem Summary begonnen. Ich brauchte nicht so lange für das Summary wie geplant, da es nicht grösser sein darf als eine A4 Seite und ich das Thema gut zusammenfassen konnte. Das Ergebnis werde ich jedoch nach den Tests erneut durchgehen, da ich dieses erst dann richtig kenne.
Ich habe mit den Testfällen angefangen und meinen Buffer dazu verwendet sie zu erweitern um wieder in meinen Zeitplan zu kommen. Ich denke, dass ich spätestens nach der Verwendung der morgigen Buffer-Zeit wieder in meinem Zeitplan bin.
Ich habe heute an meinem IPA-Bericht weitergeschrieben und erklärt, wieso ich mich für die Projektmanagement Methode IPERKA entschieden habe.

Next Steps:

Ich werde weitere Testfälle schreiben und meine bisherigen Dokumente erneut durchlesen und korrigieren

Tag 7

Datum		28.03.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Michael Speckien, Anton Kryenbuehl		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
4.3.3	Testfälle schreiben/anpassen	Abgeschlossen	1.0	1.0
5.1.1	Softwaredokumentation kontrollieren	Abgeschlossen	1.0	1.0
5.2.1	Tests durchführen	In Arbeit	3.5	2.5
7.3	Ist-Zeitplan	In Arbeit	0.5	0.5
7.4	Arbeitsjournal	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	0.5	0.5
Besprechung		**Personen**		
Statusmeeting		Michael Speckien		
Tester Meeting		Anton Kryenbuehl		

Tätigkeiten:

Ich habe mit dem Schreiben der Testfälle abgeschlossen. Dann ist mir ein Problem aufgefallen. Die Diagrammfunktion hatten einen Fehler, welche das Programm immer wieder abstürzen liess. Ich ermittelte den Fehler schnell, und erkannte, dass es in Office Versionen unter 2010 «SeriesCollection()» und nicht «FullSeriesCollection()» heisst. Anton Kryenbuehl und ich hatten daraufhin ein Meeting und schauten uns gemeinsam die Useability von meinem Programm auf meinem Laptop an. Den Fehler versuche ich in der Fehlerbehebung zu lösen. Er bewertete dies nach den Fragen, die ich mir im Vorfeld notiert habe und ihm abgegeben habe. Er hat mir ein Feedback gegeben, welches ich mir notiert habe und bei der Fehlerverbesserung beheben werde. Ich habe meine Tests ausgeführt und dabei ein paar Fehler entdeckt. Ich bin noch nicht ganz fertig mit dem Ausführen der Tests, ich werde dies morgen weiterführen. Ich werde auch diese Fehler in der Fehlerverbesserung verbessern und im Nachtest nochmals prüfen.
Ich habe heute wieder ein wenig Zeit in meinen IPA-Bericht gesteckt. Ich habe weiter dokumentiert und habe unter anderem alle Tasks im Zeitplan genau erklärt, damit klar ist, was gemeint ist. Ich werde morgen und nächste Woche noch genug Zeit haben um meine IPA-Dokumentation weiter zu bearbeiten. Um wieder in den Zeitplan zu kommen habe ich meinen Buffer von heute verwendet.

Besprechungen:

Ich hatte mit Michael heute wieder ein Statusmeeting. Wir sind das Feedback von Anton durchgegangen. Wir haben die Testfälle angeschaut und wir haben uns darauf geeinigt, dass die Anzahl Tests gut ist. Wir haben die nächsten Schritte besprochen und ich ging wieder an die Arbeit.

Im Meeting mit dem Tester hat Anton Kryenbuehl mein Programm ohne weitere Anleitung ausgeführt und getestet ob er damit klarkommt. Ich habe ihm ein Dokument mit Fragen überreicht, die er mir sorgfältig beantwortete. Im Grossen und Ganzen fand er das Programm leicht bedienbar.

Next Steps:

Ich werde morgen mit dem Testen weiterfahren. Nach den letzten Tests kommt die Fehlerbehebung. In der Fehlerbehebung versuche ich so viele Fehler wie möglich zu verbessern um dem Programm den letzten Feinschliff zu geben. Nach der Verbesserung werden die Nachtests gemacht.

Arbeitsjournal Tag 8

Datum		29.03.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Keine		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
5.2.1	Tests durchführen	Abgeschlossen	3.5	3.5
5.2.2	Fehlerbehebung Tests	Abgeschlossen	2.0	2.0
5.2.3	Nachtests durchführen	Abgeschlossen	2.0	2.0
6.1.1	Tests für Schlussbericht auswerten	In Arbeit	0.5	0.5
7.3	Arbeitsjournal	In Arbeit	0.5	0.5
7.4	Ist-Zeitplan	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	0.5	0.5
Besprechung		**Personen**		
Keine		Niemand		

Tätigkeiten:

Ich habe am Morgen mit dem testen weiter gemacht. Ich habe sie abgeschlossen und mit der Fehlerbehebung weiter gemacht. Ich habe geschafft viele Fehler und Wünsche zu verbessern oder implementieren. Ich musste dabei priorisieren was zuerst behoben werden muss. Eine Entscheidung die ich gefällt habe ist, dass das Programm für frühere Office Versionen verfügbar sein soll. Etwas Neues das ich hinzugefügt habe ist die Funktion, dass alte Trends automatisch im UI geladen werden. Wenn man nun etwas Falsches eingegeben hat, kann man dies durch einen erneuten Klick auf den Trend Button im Overview beheben.
Danach habe ich einen Nachtest ausgeführt, bei dem ich gesehen habe, dass das meiste behoben wurde.
Ich habe am Schluss alle Tests für den Schlussbericht angefangen auszuwerten. Ich werde nächste Woche dafür noch Zeit haben und es nächste Woche erledigen.

Next Steps:

Als nächstes werde ich die Tests fertig auswerten und den Schlussbericht verfassen. Ich habe am Dienstag ein letztes Statusmeeting vor der Abgabe. Am Mittwochabend werde ich meine IPA abgeben.

Tag 9

Datum		03.04.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Michael Speckien		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
6.1.1	Tests für Schlussbericht auswerten	Abgeschlossen	1.0	1.0
6.2.1	Schlussbericht verfassen	Abgeschlossen	1.0	1.0
6.2.2	Rework Flussdiagramm /Planungsdiagramme	Abgeschlossen	1.0	1.0
7.3	Ist-Zeitplan	In Arbeit	0.5	0.5
7.4	Arbeitsjournal	In Arbeit	0.5	0.5
7.5	IPA-Bericht	In Arbeit	2.0	2.0
Besprechung		**Personen**		
Statusmeeting		Michael Speckien		

Tätigkeiten:

Heute beginnt die letzte Woche meiner IPA. Ich bin teils erleichtert, teils aber auch nervös. Ich habe heute die Tests fertig ausgewertet und den Schlussbericht verfasst. Ich habe im Schlussbericht geschrieben, dass ich im Grossen und Ganzen zufrieden mit meiner Leistung bin. Ich habe ausserdem weiter am IPA-Bericht gearbeitet, da dieser auch fertig werden muss. Es gibt nicht mehr viel zu tun. Ich habe heute ausserdem eine erste Sicherheitsversion hochgeladen, um sicher zu gehen, dass der Upload Prozess klar ist und reibungslos abläuft. Dies gelang ohne weitere Probleme.

Besprechungen:

Ich hatte heute das letzte Statusmeeting mit Michael Speckien vor der Abgabe. Wir haben über die nächsten und letzten Schritte gesprochen. Wir haben beschlossen, dass falls Funktionen bei der IPA Abgabe nicht perfekt funktionieren, ich sie nach der IPA verbessere.

Next Steps:

Ich werde morgen die letzten Feinschliffe und unternehmen. Ich habe den IPA-Bericht noch nicht ganz fertig, brauche aber vermutlich nicht mehr sehr lange. Ich sollte pünktlich abgeben können.

Tag 10

Datum		04.04.2018		
Status		Im Zeitplan		
Hilfestellung		Keine		
Mailverkehr		Kein		
Besprechungen		Keine		
ID	**Tätigkeit**	**Status**	**Soll(h)**	**Ist(h)**
7.3	Ist Zeitplan führen	Abgeschlossen	0.5	0.5
7.4	Arbeitsjournal	Abgeschlossen	0.5	0.5
7.5	IPA-Bericht	Abgeschlossen	5.0	5.0
7.6	Ist-Zeitplan	In Arbeit	1.5	1.5
Besprechung		**Personen**		

Tätigkeiten:

Heute habe ich den letzten Teil meiner IPA fertig gestellt. Ich habe die Arbeitsjournale fertig gestellt und den IPA-Bericht fertig geschrieben. Im Grossen und Ganzen bin ich zufrieden mit meinem Resultat.

Next Steps:

Ich werde nun für meinen Fachvorgesetzten ein Exemplar des IPA-Berichtes ausdrucken und meiner Hauptexpertin senden. Ich werde ihn per Email versenden, da ich mir nicht sicher bin ob sie noch in China ist. Ich werde ausserdem alles auf das PkOrg hochladen. Beim Zwischengespräch mit meiner Hauptexpertin wurde mir gesagt, dass ich für keine der Experten eine weitere Dokumentation ausdrucken müsse, sondern dass der Upload auf das PkOrg genügt. Weiterhin werde ich mich für die Präsentation vorbereiten und die Funktionalitäten weiter erweitern.

2. Teil 2

2.1 IPA Kurzfassung

2.1.1 Ausgangssituation

Das TsNetV2 ist ein Testautomationstool mit dem man Geräte testen kann. Die Daten, welche man in solch einem Test hat, bestehen aus vielen Zahlen, Bereichen und Booleans. Diese so auszulesen ist sehr mühsam und unübersichtlich. Ziel der IPA ist es, diese Daten grafisch in einem bis mehreren Diagrammen darstellen zu lassen. Die Booleans sollen dabei im eigentlichen Minus Bereich angezeigt werden. Die Visualisierung soll helfen die Daten schneller auslesen und verstehen zu können.

2.1.2 Umsetzung

Die Anforderungen, welche in der Ausgangssituation genannt wurden, werden im Rahmen der individuellen praktischen Arbeit umgesetzt.

In der Informationsphase habe ich alle wichtigen Informationen gesammelt, welche ich für die Umsetzung brauchte.

In der Planungsphase habe ich einen Zeitplan erstellt, auf dem die absolute Zeitachse, also 10 Tage, definiert ist. Der Zeitplan wurde nach der Projektmanagement Methode IPERKA aufgebaut. Das bedeutet, dass ich die Tätigkeiten in die 6 Phasen aufgeteilt habe. Zusätzlich habe ich die Aufgaben nochmals in 3 Stufen, UI, Programm und Dokumentation aufgeteilt. Ich habe viel Zeit ins UI investiert, da dies meiner Meinung nach wichtig für die spätere Umsetzung war. Ich habe Struktogramme und Flussdiagramme erstellt. Durch die Struktogramme und der genauen Aufgabenbeschreibung konnte ich später, in der Realisierungsphase alles umsetzen.

Für die Realisierung musste ein UI erstellt werden. Weiterhin musste die Generierung der Hilfstabelle und die Diagrammerstellung implementiert werden. Die Daten die in die Hilfstabelle gespeichert werden, werden den Einstellungen entsprechend abgeändert und bei der Diagrammerstellung angezeigt.

In der Kontrollphase habe ich mein Programm, mit den davor aufgeschriebenen Akzeptanztests getestet. Ein Siemens Mitarbeiter hat Useability Tests durchgeführt, woran man erkennen konnte, was am Programm noch nicht ganz selbstverständlich für den User ist.

2.1.3 Ergebnis

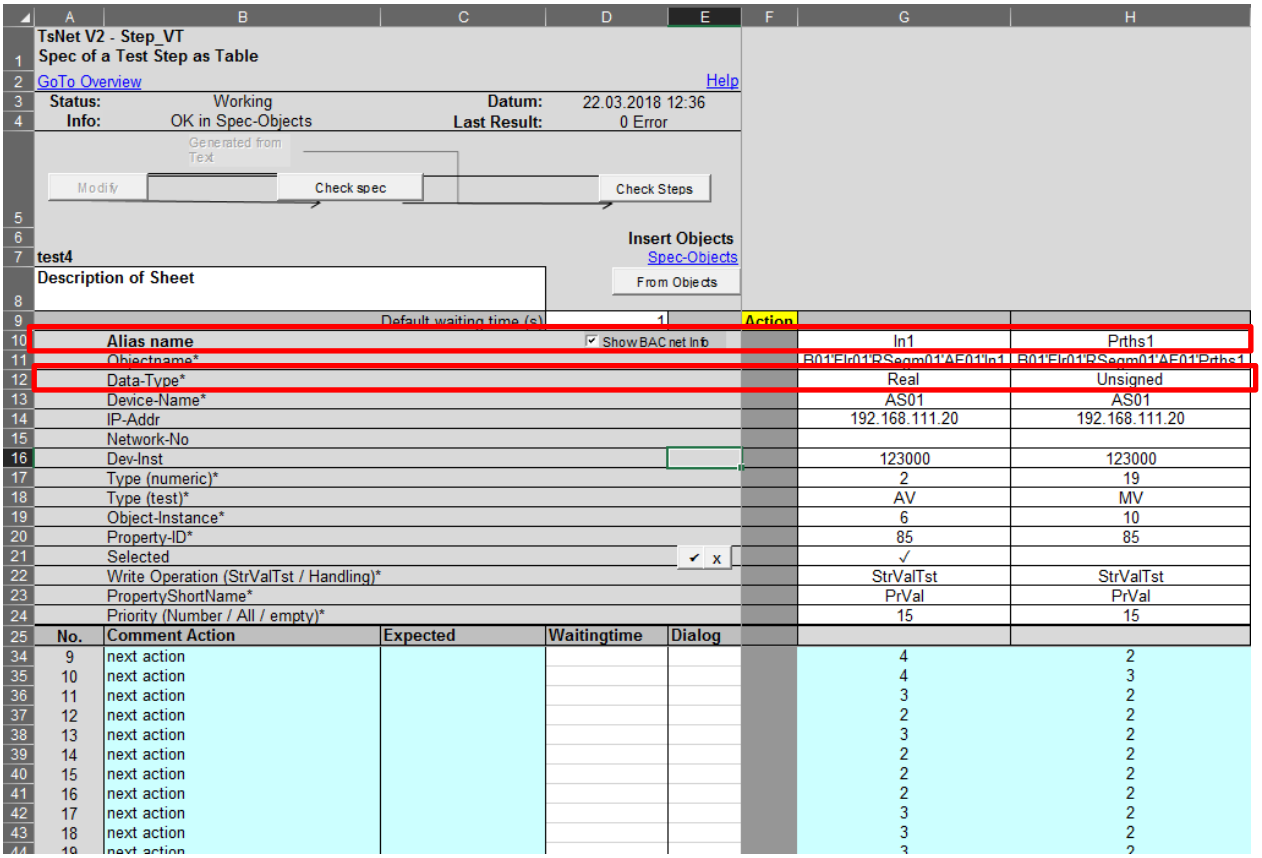
Das Produkt wurde gemäss Aufgabenstellung im PkOrg gelöst. Es konnte alles implementiert werden und das Produkt könnte so in den Betrieb aufgenommen werden. Es war möglich, zusätzliche Funktionen zu implementieren, welche dem User den Umgang mit dem UI vereinfacht.

2.2 Realisierung

2.2.1 Umfeld

Das Umfeld ist das TsNet V2. Das TsNet besteht aus 2 Teilen. Einem Definitionsteil und einem Run-Time-Teil. Es ist ein Testautomationstool, welches Controller testet. Der Definitionsteil wurde in den letzten Jahren erneuert, weshalb ich schon einen Einblick ins TsNet bekommen habe. Der Definitionsteil dient zur Spezifizierung der Tests und enthält die Werte welches ich im Diagramm darstellen möchte. Ich liste folgend einige Sheets auf, die für meine IPA relevant waren.

Die Bedienoberfläche ist grundsätzlich das TsNetV2. Im TsNetV2 befinden sich die Tests, welche in einem Diagramm dargestellt werden sollen.



TsNet V2 - Step_VT
Spec of a Test Step as Table
GoTo Overview | Help
Status: Working | Datum: 22.03.2018 12:36
Info: OK in Spec-Objects | Last Result: 0 Error
Generated from Text
Modify | Check spec | Check Steps
Insert Objects
Spec-Objects
From Objects
test4
Description of Sheet
Default waiting time (s): 1
Show BACnet Info
Action

10	Alias name				In1	Prths1
11	Objectname*				B01'Flr01'RSegm01'AF01'In1	B01'Flr01'RSegm01'AF01'Prths1
12	Data-Type*				Real	Unsigned
13	Device-Name*				AS01	AS01
14	IP-Addr				192.168.111.20	192.168.111.20
15	Network-No					
16	Dev-Inst				123000	123000
17	Type (numeric)*				2	19
18	Type (test)*				AV	MV
19	Object-Instance*				6	10
20	Property-ID*				85	85
21	Selected			✓ x	✓	
22	Write Operation (StrValTst / Handling)*				StrValTst	StrValTst
23	PropertyShortName*				PrVal	PrVal
24	Priority (Number / All / empty)*				15	15
25	No.	Comment Action	Expected	Waitingtime	Dialog	
34	9	next action				4	2
35	10	next action				4	3
36	11	next action				3	2
37	12	next action				2	2
38	13	next action				3	2
39	14	next action				2	2
40	15	next action				2	2
41	16	next action				2	2
42	17	next action				3	2
43	18	next action				3	2

Abbildung 10: TsNet Test

Die für die Diagrammdarstellung relevanten Teile des Tests sind die «Alias names» als Objektnamen, «Data-Type» als Datentypen und die Werte des Tests.

In der selben Datei befindet sich das Tabellenblatt Overview. Im Overview befindet sich der Button «Trend» welcher meine Funktionen startet. Die Funktionen werden später in diesem Dokument genauer erläutert.

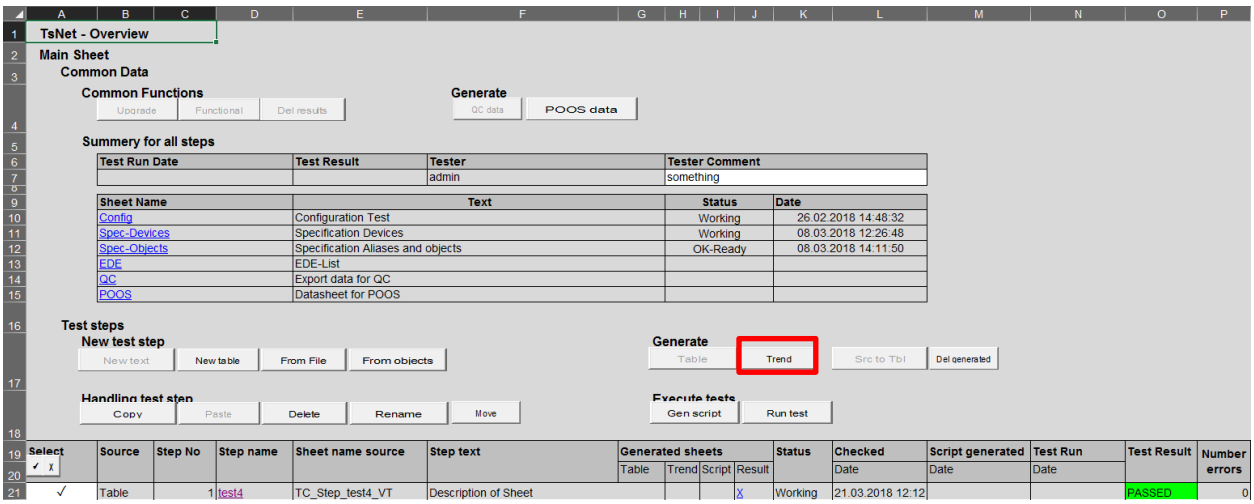


Abbildung 11: TsNet Overview

Die für die Diagrammdarstellung relevanten Teile des Overviews ist die Selektion der Tests und der Button «Trend». Ab diesem Button startet meine Aufgabe.

2.2.2 Bedienoberfläche / UI

Die Bedienoberfläche wurde von mir erstellt. Sie wird als erstes aufgerufen, wenn man auf den Button «Trend» klickt. Dies ist die Bedienoberfläche für die Diagrammerstellung an sich.

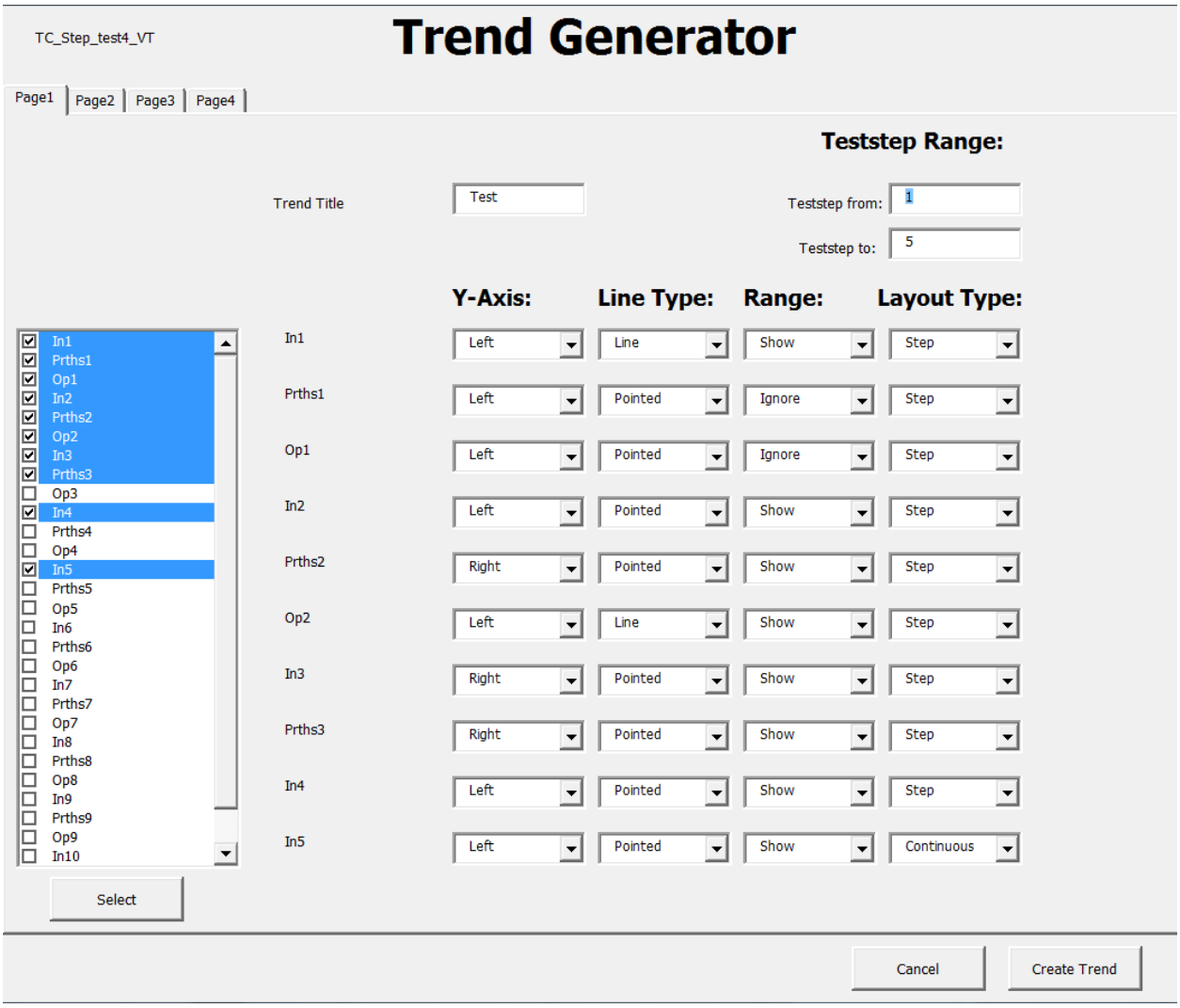


Abbildung 12: User Interface

Das UI besteht aus der Objektanzeige und den Einstellungen.

Objektanzeige

Die Objektanzeige wurde in Form eines «ListBox» Steuerelement realisiert. Es enthält alle möglichen Objekte aus dem selektierten Test. Alle möglichen Objekte bedeutet, alle Objekte welche nicht Struct, Array oder String als Datentyp haben. Jedes Objekt wird mit einem «CheckBox» Steuerelement versehen um sie selektieren zu können.

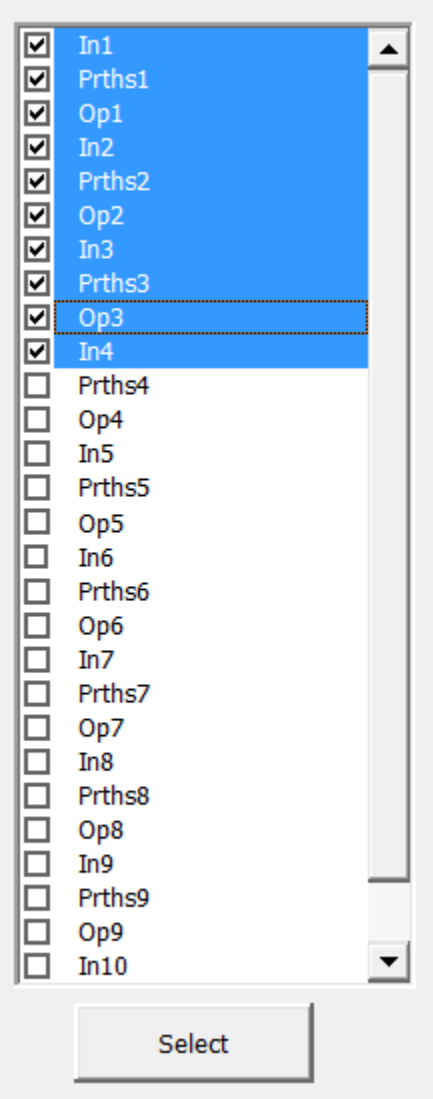


Abbildung 13: Objektanzeige

Einstellungen:

Der User kann insgesamt 4 Diagramme pro Test erstellen. Pro Diagramm sind maximal 10 mögliche Objekte zuweisbar. Pro Objekt hat man 4 verschiedene Einstellungsmöglichkeiten.

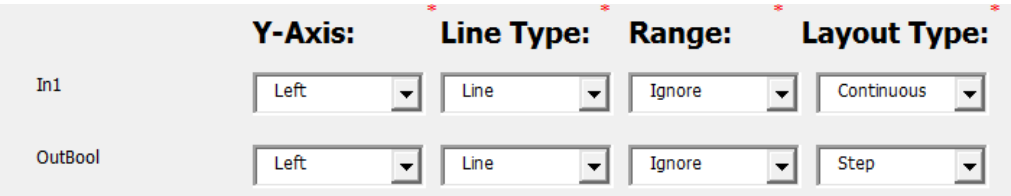


Abbildung 14: Einstellungen

Y-Axis:

Bei dieser Einstellung kann man bestimmen, ob das Objekt auf der linken Y-Achse, also die Primärachse, dargestellt werden soll oder ob das Objekt auf der rechten Y-Achse, also die Sekundärachse, dargestellt werden soll.

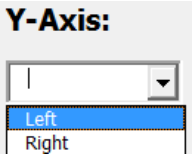


Abbildung 15: Einstellung Y-Axis

Line Type:

Bei dieser Einstellung kann man bestimmen, ob das Objekt im Diagramm mit einer gepunkteten oder durchgezogenen Linie dargestellt werden soll.

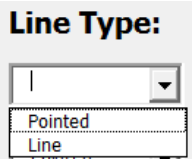


Abbildung 16: Einstellung LineType

Range:

Manchmal kommt es vor, dass in einem Test ein Bereich eingetragen ist z.B. 2..25. Mithilfe dieser Einstellung kann man entscheiden, wie man damit Umgehen möchte. Man kann den Bereich entweder anzeigen oder nicht anzeigen. Wenn man den Bereich nicht anzeigen möchte, wird einfach der kleinere Wert von beiden genommen, wenn der Bereich angezeigt werden soll, wird im Diagramm eine Linie generiert, welche den kleineren Wert von beiden anzeigt und der grössere Wert wird mit einer gepunkteten Linie darüber angezeigt.

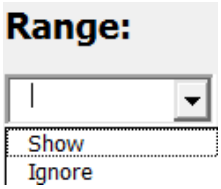


Abbildung 17: Einstellung Range

Type:

Mit dieser Einstellung können wir festlegen, wie das Objekt dargestellt werden soll. Man kann entweder «Continuous» angeben, was bedeutet, dass die Linie interpoliert wird, oder man gibt «Step» an, was bedeutet, dass die Linie in einer Art «Treppendarstellung» angezeigt wird.

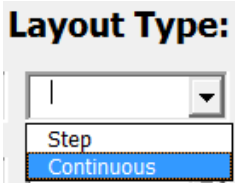


Abbildung 18: Einstellung Type

Boolean:

Falls das Objekt ein Boolean ist, werden die Einstellungsmöglichkeiten Y-Axis und Type blockiert und die Standardwerte «Left» und «Step» hineingeschrieben.

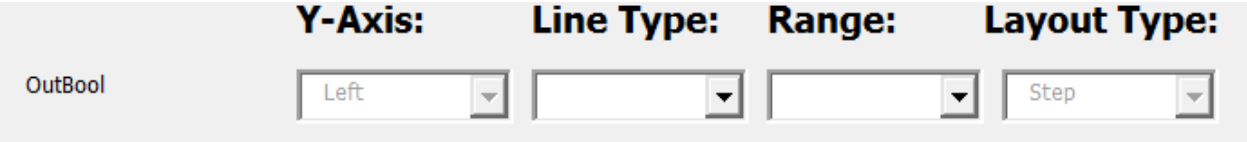


Abbildung 19: Boolean Sperre

Die Erstellung der drei anderen Diagramme:

Wie bereits erwähnt, kann man bis zu 4 Diagramme erstellen. Deshalb besitzt das UI das «Multiseiten» Steuerungselement. Dies ermöglicht die 4 Diagramme auf einmal erstellen zu können, welche bis auf den Test unabhängig voneinander sind.

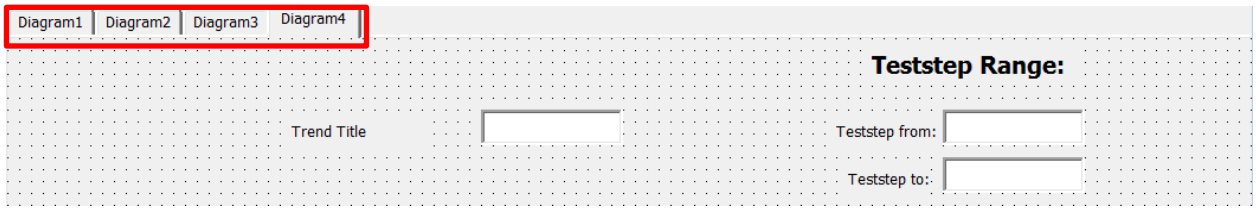


Abbildung 20: Multi Seiten

Benennung

Um den Code dynamisch gestalten zu können ist eine Benennung der Steuerelemente sehr wichtig. Ich habe sie folgendermassen benannt:

Steuerelement + Einstellung + Seite + Objektnummer

Als kleines Beispiel:

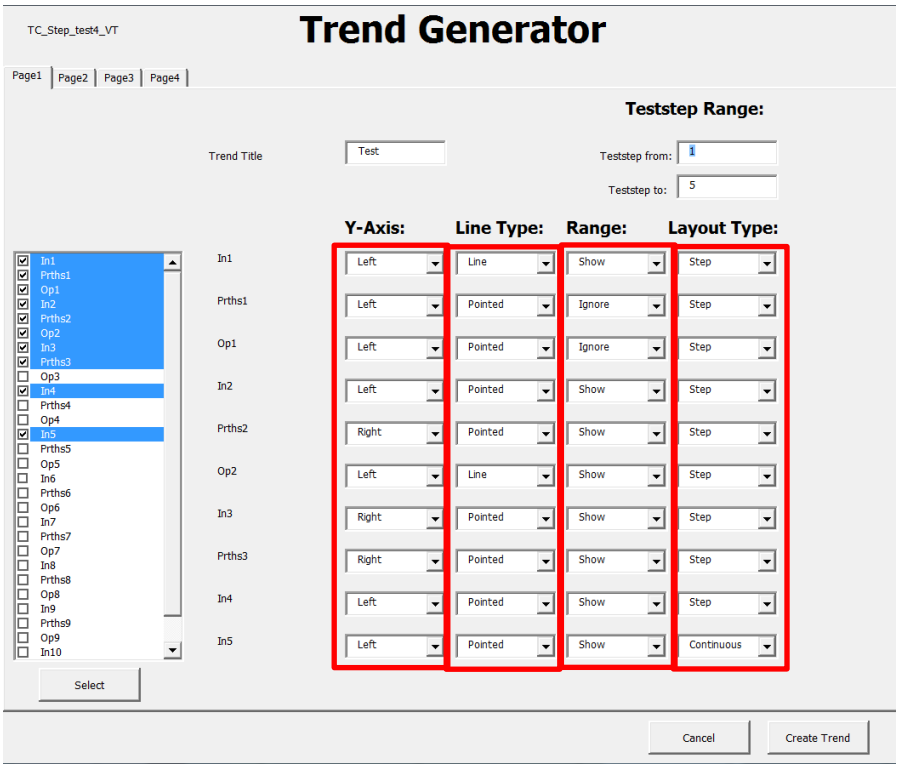


Abbildung 21: UI Beschriftung

Bei diesem Bild heissen die markierten Elemente:

ComboBoxYAxis10 – 19

ComboBoxLineType 10 – 19

ComboBoxRange 10 – 19

ComboBoxType 10 – 19

2.2.3 Hilfstabelle

Die Hilfstabelle wird während der Erstellung für das Diagramm generiert. In der Hilfstabelle stehen alle selektierten Objekte, wie auch die Einstellungen dazu. Unter den Einstellungen und Objektnamen werden die Werte des Testes eingetragen. Jeder Wert wird 5-mal eingetragen. Dies hat mit der Darstellung zu tun.

9			Default waiting time (s)	1		Action			
10		Alias name			Show BACnet Info		In1	Prths1	Op1
11		Objectname*					B01'Flr01'RSegm01'AF01'In1	B01'Flr01'RSegm01'AF01'Prths1	B01'Flr01'RSegm01'AF01'Op1
12		Data-Type*					Real	Unsigned	Unsigned
13		Device-Name*					AS01	AS01	AS01
14		IP-Addr					192.168.111.20	192.168.111.20	192.168.111.20
15		Network-No							
16		Dev-Inst					123000	123000	123000
17		Type (numeric)*					2	19	19
18		Type (test)*					AV	MV	MV
19		Object-Instance*					6	10	11
20		Property-ID*					85	85	85
21		Selected			✓ x				
22		Write Operation (StrValTst / Handling)*					StrValTst	StrValTst	StrValTst
23		PropertyShortName*					PrVal	PrVal	PrVal
24		Priority (Number / All / empty)*					15	15	15
25	No.	Comment Action	Expected	Waitingtime	Dialog				
26	1	Start of Tests					0	3	4
27	2	next action					0	3	1
28	3	next action					2..7	2	2..7
29	4	next action					1	3	2

Abbildung 22: TsNet Test

	A	B	C	D	E	F	G	H	I	J	K	L
1	Diagramm 1		Test Sheet:	;_Step_Test5_	From Teststep	1	To TestStep:	28		Create Trend	Modify	
2	LineType	Line	Line	Line	Pointed							
3	Range	Ignore	Show	Ignore	Ignore							
4	Type	Step	Step	Continuous	Step							
5	Y-Axis	Left	Left	Right	Left							
6	No	In1	Prths1	Op1-->	OutBool							FALSCH
7		0	3	4	-2.8							-2.8
8		0	3	4	-2.8							-2.8
9	test0	0	3	4	-2.8							-2.8
10		0	3	3.4	-2.8							-2.8
11		0	3	2.8	-2.8							-2.8

Abbildung 23: Hilfstabelle

Die Werte können je nach Einstellungen verändert werden.

Range in der Hilfstabelle

Wenn der User die Einstellung «Show» bei Range angegeben hat, wird zuerst die kleinere Zahl in die Hilfstabelle eingetragen. Danach wird eine zusätzliche Spalte mit der Überschrift «Range» erstellt und die höhere Zahl wird in die neue Spalte in der selben Zeile eingeschrieben.

Boolean in der Hilfstabelle

Falls analoge Werte in der Tabelle stehen würden, die bis z.B. 1000 gehen, nützt es nicht viel, wenn man einen Boolean in die Hilfstabelle einträgt. Booleans haben die Werte 0-1. Da die Linie so klein wäre, wäre kaum zu erkennen ob der Boolean nun True oder False ist. Deshalb wird der Boolean in der Hilfstabelle mit einer speziellen Formel bearbeitet und in den Minus Bereich gezogen.

OutBool									Range	FALSCH
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-2.8										-2.8
-1.4									2	-2.8
-1.4									7	-2.8
-1.4									7	-2.8
-1.4									7	-2.8
-1.4									2	-2.8
-1.4										-2.8
-1.4										-2.8
-1.4										-2.8

Abbildung 24: Boolean in Hilfstabelle

Continuous in der Hilfstabelle

Ist die Einstellung Continuous gewählt, wird der Wert in der Hilfstabelle ebenfalls mit einer Formel bearbeitet und die einzelnen Daten verändern sich. Nach diesen Veränderungen sollte das Objekt im Continuous Modus in dem Diagramm angezeigt werden.

2.2.4 Diagramm

Das Diagramm wird aus der Hilfstabelle generiert. Das Programm soll das Ziel erfüllen, eine einfache grafische Anzeige der Daten in einem Test anzuzeigen. Das Programm muss dabei bestimmte Einstellungen, die der User angeben kann, umsetzen und anzeigen. Dabei muss beachtet werden, dass sich die Boolean Objekte nicht im Positiven, sondern ungewöhnlicherweise im negativen Bereich befindet. Sie befinden sich im negativen Bereich, weil die Werte angepasst werden müssen um alle übersichtlich dargestellt werden zu können. Hat man z.B. einen analogen Wert, der bis 1500 geht, ist es nicht oder nur sehr schwer möglich den Boolean überhaupt zu erkennen. Die Formel dazu wird später in diesem Dokument erwähnt.

	0
	0
test0	0
	0
	0
	0
	0
test1	0
	0.4
	0.8
	1.2
	1.6
test2	2
	1.8
	1.6

Abbildung 25: Hilfstabelle mit TestSteps und Continuous Einstellung

Die X-Achse der Diagramme besteht aus den Tests links.

2.2.5 Funktionen

Nachfolgend beschreibe ich alle Funktionen, die ich für mein Projekt implementiert habe. Viele dieser Funktionen, sind Funktionen, die einen gewissen Wert zurückgeben. Ich werde zuerst die Funktionen beschreiben, welche den Trend an sich generieren und anzeigen und danach alle Funktionen, welche die benötigten Werte zurückgeben.

TrendCreatorLoad()

Die Funktion «TrendCreatorLoad()» wird durch das bestätigen des Buttons im Overview ausgeführt. Es wird überprüft, ob ein Test im Sheet Overview selektiert wurde. Der Name des selektierten Sheets wird auf eine Variable «strSheetName» gespeichert. Der Code wechselt zum ausgewählten Sheet und fügt jedes Objekt in die vier Listboxen des UI, bei denen der Datentyp nicht String, Struct oder Array ist. Diese drei Datentypen kann man in einem Diagramm nicht anzeigen und werden deshalb ignoriert. Das UI wird aufgerufen und die verfügbaren Objekte werden angezeigt. Der Name des selektierten Sheets wird ebenfalls eingetragen.

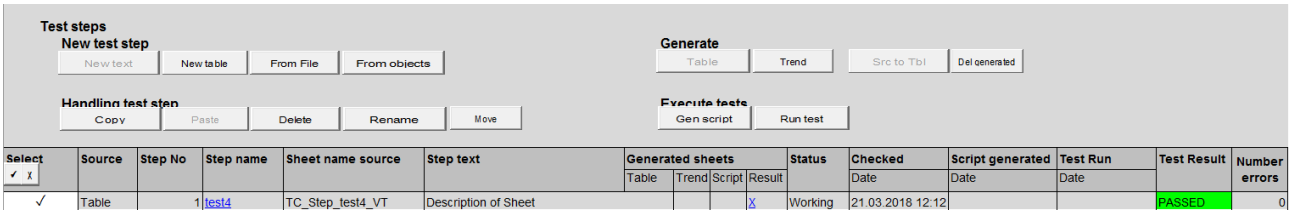


Abbildung 26: Testselektion

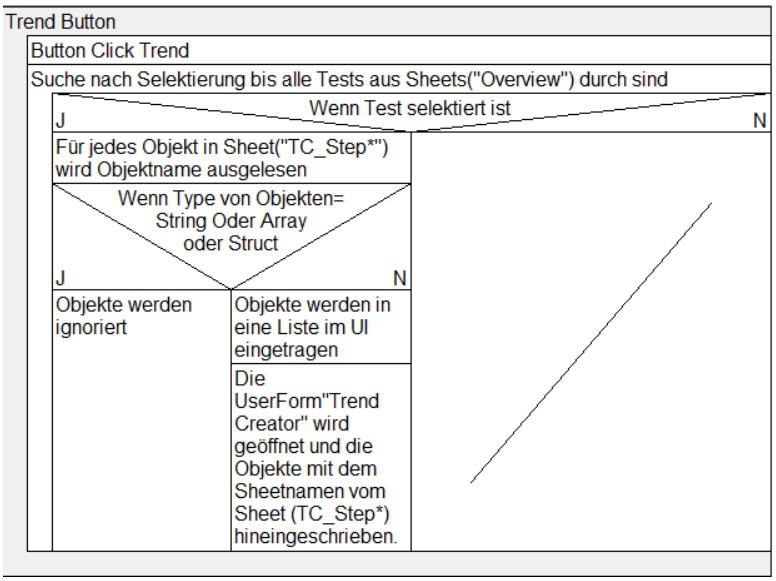


Abbildung 27: Struktogramm Trend Button

UserForm_Initialize()

Diese Funktion wird aufgerufen, sobald die Objekte zu den Listboxen hinzugefügt wurden. In dieser Funktion wird das UI hauptsächlich geladen. Durch die Einstellungsmöglichkeiten kann man das Diagramm den Wünschen entsprechend anpassen. Diese werden in dieser Funktion den Comboboxen zugewiesen und eingetragen. Die Comboboxen und alle anderen Steuerelemente ausser dem Select Button werden dann blockiert.

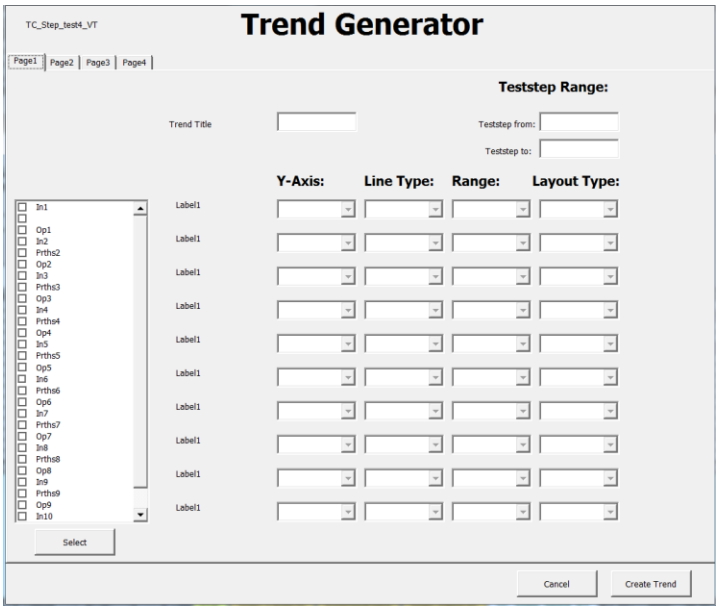


Abbildung 28: Steuerelemente Sperrung

ButtonSelect10_Click()

Diese Funktion gibt es für jede der 4 Seiten. In dieser Funktion werden die 10 ausgewählten Objekte in die rechte Spalte geschrieben. Die Comboboxen werden freigegeben, dass der User die Einstellungen pro Objekt vornehmen kann. Die Textbox für die Überschrift des Diagrammes (ist gleichzeitig der Name der Hilfstabelle) wird freigegeben. Die 2 Textboxen für die Eingabe der Teststeps werden auch freigegeben und mit einem Standard Wert aufgefüllt.

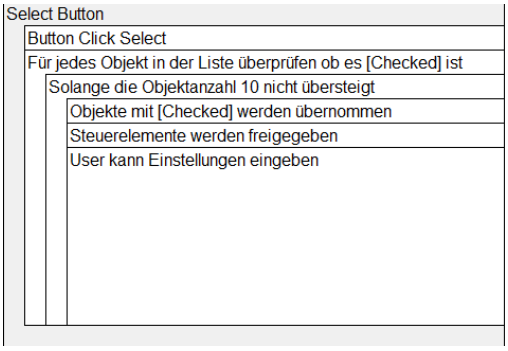


Abbildung 29: Struktogramm Select Button

CreateHelpTable()

Diese Funktion wird aufgerufen, wenn der Button «Create Trend» gedrückt wird. Der ButtonCreate_Click() ruft diese Funktion auf. Diese Funktion ist die Hauptfunktion in meinem Projekt. Zuerst zählt die Funktion wie viele Diagramme zu erstellen sind und schreibt alle Werte, die der User im UI angegeben hat in ein Array.

Als nächstes wird die Hilfstabelle erstellt und benannt. Die Einstellungen und die anderen User Eingaben werden durch das oben erwähnte Array eigespeichert.

Die Funktion wechselt erneut in das TestCase Sheet und sucht in der Reihe «Aliasname» nach den selektierten Objekten. Wenn Sie gefunden wurden, werden die Daten, welche weiter unten im Sheet stehen ausgelesen und in ein anderes Array gespeichert. Falls ein Test nicht gefunden werden sollte, wird eine Fehlermeldung ausgegeben. Diese Daten werden in die Hilfstabelle gespeichert. Jeder Wert wird 5-mal eingetragen, da dies die Diagramm Anzeige verschönert. Bei jedem 3. Eintrag des Wertes in der 2. Spalte wird in der 1. Spalte ein «Test» hinzugefügt. Auch dies dient der Darstellung.

Nachdem die Daten hinzugefügt wurden, werden die Einstellungen gelesen und die Daten entsprechend abgeändert. Je nach Einstellung werden verschiedene Funktionen aufgerufen, die für die Änderung der Daten zuständig sind oder Werte zurückgeben, welche wir weiter verrechnen müssen.

- Wenn die Einstellung «Continuous» für den Type gewählt wurde, wird TypeContinuous() aufgerufen
- Wenn die Einstellung «Show» für Range gewählt wurde, wird RangeShow() aufgerufen

Ich werde auf die Funktionen gleich weiter eingehen. Es wird weiterhin überprüft, ob es sich um einen Boolean handelt. Falls es sich um einen Boolean handelt, wird er in ein Array eingespeichert. Wenn es sich nicht um keinen Boolean handelt, werden die Funktionen

- GetMinValue()
- GetMaxValue()

aufgerufen. Ich werde auch auf diese Funktionen später zurückkommen. Nun wird der Boolean Wert manipuliert, damit er im Diagramm richtig angezeigt wird. Eine zusätzliche Spalte mit der Überschrift «False» wird generiert und die manipulierten Werte für 0 darunter eingetragen. Die Funktionen für die Manipulation heissen:

- GetFalseValue()
- GetTrueValue()

Nun kann das Diagramm erstellt werden und TrendCreation() wird aufgerufen.

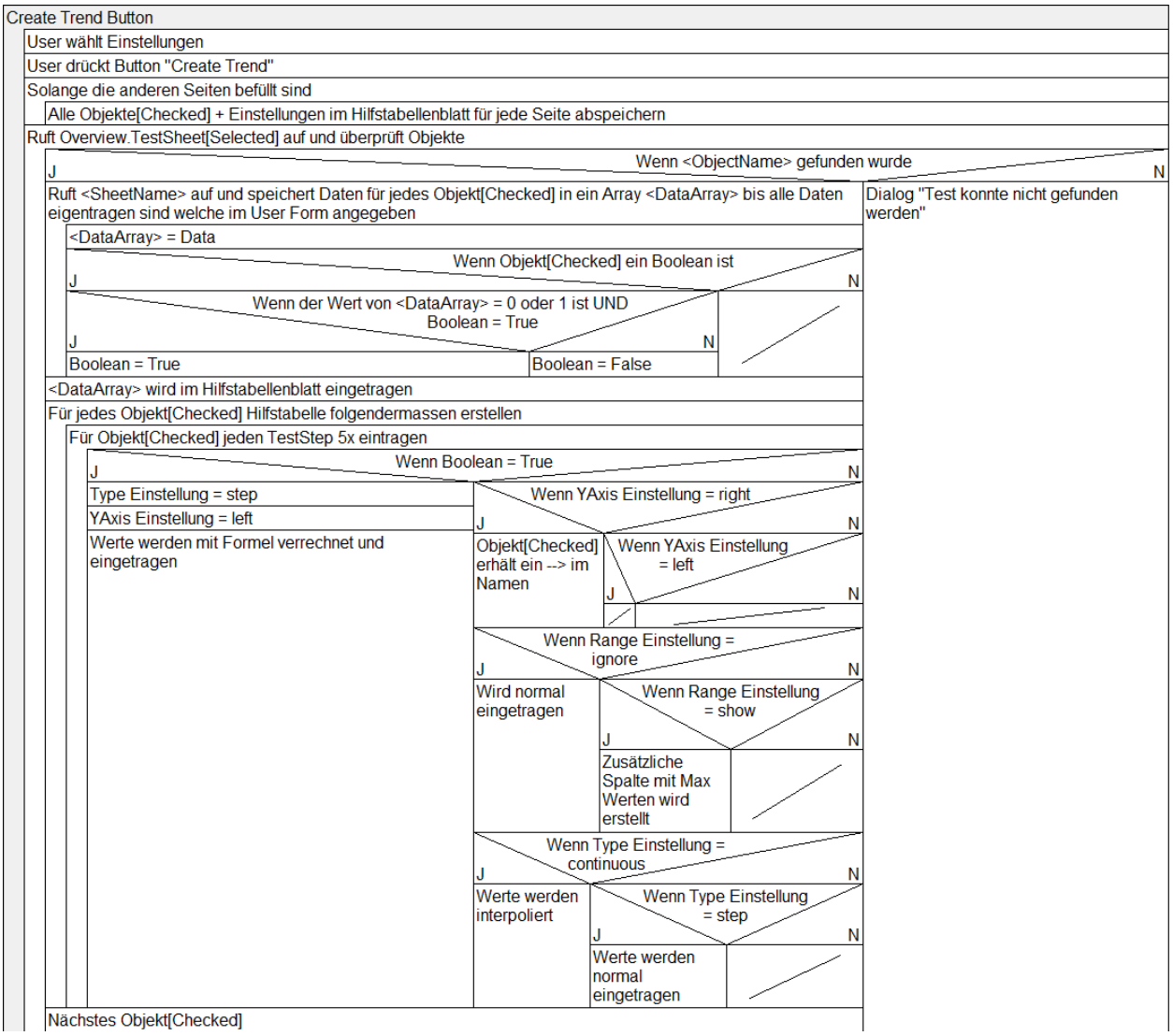


Abbildung 30: Struktogramm Trend Button/Diagrammerstellung Teil1

TrendCreation()

Diese Funktion wird am Schluss des CreateHelpTable () aufgerufen. In dieser Funktion geht es um die Diagrammerstellung. Zuerst wird das Diagramm erstellt. (dafür wird ein extra Sheet gemacht, indem bis zu 4 Diagramme angezeigt werden können.) Dem Diagramm wird eine Range zugeteilt, um die Daten dafür festzulegen. Die Einstellungen «YAxis» und «LineType» werden abgefragt. Mit der YAxis Einstellung wird bestimmt, ob ein Objekt links, an der Primärachse oder rechts, an der Sekundärachse befestigt wird. Mit der LineType Einstellung kann festgelegt werden, ob die Linie für das Objekt gepunktet oder als ganze Linie angezeigt wird. Danach wird überprüft, ob sich noch irgendwelche «Datenreihen» darin im Diagramm befinden, denn die Datenreihen sollen die Objektnamen anzeigen und nicht «Datenreihen» anzeigen. Die X-Achse bekommt übernimmt die TestSteps als X-Achsen-Wert und die Range und False Daten, welche zusätzlich angezeigt werden ins Diagramm eingefügt.

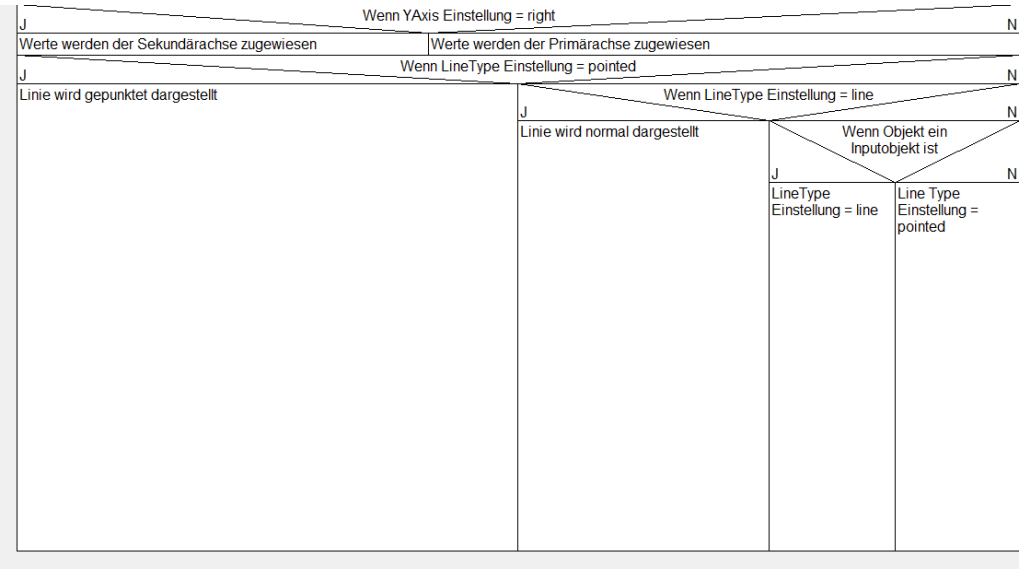


Abbildung 31:Struktogramm Trend Button/Diagrammerstellung Teil2

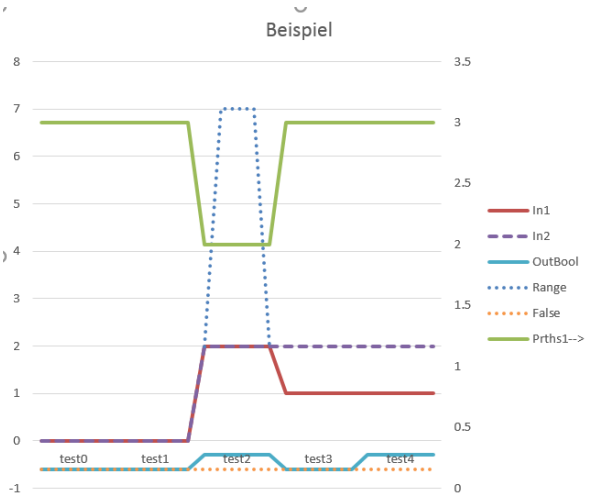


Abbildung 32: Beispiel Diagramm

Jetzt werden die Funktionen beschrieben, welche ich bei CreateHelpTable() erwähnt habe. Sie sind nicht ganz so komplex wie die bisherigen Funktionen, jedoch aber sehr wichtig, da von ihnen die Berechnung durchgeführt wird.

TypeContinuous()

TypeContinuous() wird im CreateHelpTable() aufgerufen. Diese Funktion wird aufgerufen, wenn eine oder mehrere Einstellungen den Wert «Continuous» enthalten. Da jeder TestStep 5 Mal in der Hilfstabelle eingetragen wird, behält der 3. Wert den Originalen Wert und die anderen 4 werden abgeändert.

Z1	Der Wert wird interpoliert
Z2	Der Wert wird interpoliert
Z3	Der Wert bleibt
Z4	Der Wert wird interpoliert
Z5	Der Wert wird interpoliert

Tabelle 8: Interpolierung

Die Werte werden dann später mit folgender Formel berechnet

Z1	Testwert – (-(Z3 des vorherigen Testschritts – Testwert) * 0.4)
Z2	Testwert – (-(Z3 des vorherigen Testschritts – Testwert) * 0.2)
Z3	Der Wert bleibt
Z4	Testwert + (Z3 des nächsten Testschritts – Testwert) * 0.2
Z5	Testwert + (Z3 des nächsten Testschritts – Testwert) * 0.4

Tabelle 9: Interpolierung Formel

Beim ersten und letzten Testschritt kann man diese Formel natürlich nicht genau so verwenden, da z.B. beim ersten Testschritt kein vorheriger Testschritt existiert. In diesem Fall behält der Wert seinen Ursprungswert. Die Werte werden dann in die Zellen in der Hilfstabelle geschrieben.

RangeShow()

RangeShow() wird im CreateHelpTable() aufgerufen. Die Funktion wird aufgerufen, wenn Range die Einstellung «Show» enthält. Wenn die Funktion aufgerufen wurde, wird der Testwert Zelle für Zelle nach dem Wert abgefragt. Wenn in einer Zelle ein «..» auftaucht, wird der Wert nach «..» gesplittet. Die 2 Werte die sich daraus ergeben werden in einem Array gespeichert. Nun wird der niedrigere Wert in die Hilfstabelle anstelle des alten Wertes eingefügt. Als nächstes wird eine neue Spalte mit der Überschrift «Range» erstellt. Auf derselben Zeile wie die Originalwerte wird nun zuerst die kleinere Zahl von beiden eingetragen, dann 3-mal die grössere und am Schluss noch einmal die Kleine um den Bereich im Diagramm sauber darzustellen.

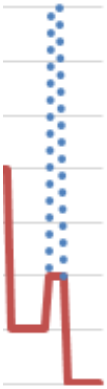


Tabelle 10: Range anzeige

GetMinValue()

Diese Funktion wird im CreateHelpTable () aufgerufen. Diese Funktion wird für die Berechnung der modifizierten Boolean Werte verwendet. Sie sucht nach dem kleinsten analogen Wert. Am Anfang wird der kleinste Wert testweise auf 1000 gesetzt. Danach werden jeweils die analogen Werte miteinander verglichen. Der kleinste Wert wird gespeichert und an CreateHelpTable() zurückgegeben.

GetMaxValue()

Diese Funktion wird im CreateHelpTable() aufgerufen. Auch diese Funktion wird für die Berechnung der modifizierten Boolean Werte verwendet. Die Funktion sucht nach dem grössten analogen Wert. Genau wie im GetMinValue() werden die Werte miteinander verglichen und der grösste Wert wird gespeichert und and CreateHelpTable() zurückgegeben.

GetFalseValue()

Diese Funktion wird im CreateHelpTable() aufgerufen, jedoch nur wenn ein Boolean im Diagramm angezeigt werden soll. Diese Funktion enthält eine Berechnung, die den Boolean so verändert, dass er in einem Diagramm besser angezeigt werden kann. Die Berechnung für den False Wert lautet:

Minimum – (Anzahl boolsche Werte * Differenz /5)

Tabelle 11: Formel False Value

Dieser Wert wird dann an CreateHelpTable() zurückgegeben. Wie oben erwähnt, wird eine neue Spalte mit der Überschrift «False» generiert. Die ganze neue Spalte wird dann mit diesem Wert aufgefüllt.

GetTrueValue()

Diese Funktion wird auch im CreateHelpTable() aufgerufen, wenn ein Boolean im Diagramm angezeigt werden soll. Diese Funktion enthält die Berechnung für die True Werte. Die Formel für die Berechnung lautet:

1 * (GetMaxValue - GetMinValue) / 10 + GetFalseValue

Tabelle 12: Formel True Value

Wie man in der Berechnung sehen kann, verwenden wir hier die GetMaxValue() und GetMinValue() Funktion. Ausserdem wird mit dem GetFalseValue() addiert, da wir den True Wert bekommen möchten.

2.2.6 Felder

Folgender Felder sind in den neu erstellten Trend Tabellenblätter definiert

Feld Namen	Zellenbereich	Beschreibung
TestSheetName	D1	In dieser Zelle befindet sich der Name des Tests, aus dem die Objekte für die Diagrammerstellung geholt werden.
ValueFromTestStep	F1	In diesem Feld befindet sich der 1. Testschritt aus den Tests, der in dem Diagramm angezeigt werden soll.
ValueToTestStep	H1	In der Zelle «ValueToTestStep» befindet sich der letzte Testschritt aus den Tests, der in dem Diagramm angezeigt werden soll.

Tabelle 13: Felder

2.2.7 Entwicklungsumgebung

Ich arbeite mit der Entwicklungsumgebung von Excel. Ich programmiere in dieser Umgebung mit VBA.

Abbildung 33: Module

2.3 Kontrollieren

2.3.1 Test Konzept

Ich habe für mein Programm verschiedene Tests gemacht. Als Testanlage habe ich dafür mein Programm in der Exceldatei gewählt. Einerseits habe ich mit Anton Kryenbuehl, ein Mitarbeiter in der Siemens, Useability Tests gemacht. Ich habe dafür ein Dokument mit ein paar Fragen vorbereitet, die er mir bestmöglich beantworten sollte. Andererseits habe ich Akzeptanztests geschrieben und sie in 2 Kategorien unterteilt. Die eine Kategorie ist «Test der Eingabedialoge». In dieser Kategorie habe ich alle Eingabedialoge getestet. Ich habe sie nicht nur auf richtige, sondern auch auf falsche Eingaben getestet um Abstürze des Programms zu vermeiden. Das Ziel ist, dass das Programm alle möglichen falschen Usereingaben abfängt.

Test Daten:

Die Test Daten befinden sich alle in der Excel Datei mit der ich gearbeitet habe. Es handelt sich um normale Tests, welche man innerhalb des TsNet erstellen kann.

Testmethoden:

Akzeptanztests wurden erstellt. Diese habe ich in 2 Kategorien eingeteilt und selber getestet. Ausserdem wurden Useability Tests gemacht. Diese hat Anton Kryenbuehl für mich getestet.

Eingabedialoge Tests

Test ID: 1.0	
Name	Eingabedialoge: From/To StepNo
Testvoraussetzung	
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User wählt Objekte und klickt auf Select. • User wählt Einstellungen für die Objekte. • Textbox für die Überschrift wird ausgefüllt. • TestSteps von 1-10 wird angegeben.
Erwartetes Resultat	TestSteps 1-10 werden nach der Einstellung verarbeitet und in die Hilfstabelle geschrieben.
Test ID: 1.1	
Name	Eingabedialoge: From/To StepNo
Testvoraussetzung	
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User wählt Objekte und klickt auf Select. • User wählt Einstellungen für die Objekte. • Textbox für die Überschrift wird ausgefüllt. • TestSteps von 10-1 wird angegeben.
Erwartetes Resultat	Eine Fehlermeldung erscheint, dass der User keine grössere Zahl als From TestStep angeben kann. Die erste Zahl muss immer kleiner sein.
Test ID: 1.2	
Name	Eingabedialoge: From/To StepNo
Testvoraussetzung	
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User wählt Objekte und klickt auf Select. • User wählt Einstellungen für die Objekte. • Textbox für die Überschrift wird ausgefüllt.

	• TestSteps wird in Buchstaben angegeben.
Erwartetes Resultat	• Fehlermeldung wird ausgegeben, da nur Zahlen in diesen Eingabefeldern erwartet werden.
Test ID: 1.3	
Name	Eingabedialoge: From/To StepNo
Testvoraussetzung	
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User wählt Objekte und klickt auf Select. • User wählt Einstellungen für die Objekte. • Textbox für die Überschrift wird ausgefüllt. • TestSteps werden nicht angegeben.
Erwartetes Resultat	Es werden standardmässig 1 – maximale Anzahl TestSteps verwendet.
Test ID: 2.0	
Name	Eingabedialoge: TextBoxTrendHeader
Testvoraussetzung	
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User gibt an, welcher Bereich der Teststeps verwendet wird. • User gibt Einstellungen ein. • Hilfstabelle wird generiert. • User klickt auf Button «Save Trend». • User schreibt in Eingabedialog einen Text für die Überschrift des Diagramms.
Erwartetes Resultat	Sheet wird abgespeichert und Diagramm wird erstellt. Das Diagramm trägt den Titel des Textes in der Texbox.
Test ID: 3.0	
Name	Eingabedialoge: Einstellungen
Testvoraussetzung	
Testablauf	• User klickt auf Trend Button • Wechsel auf UI «Trend Creator» • Objektnamen werden geladen • User wählt Objekt und klickt auf «Select»
Erwartetes Resultat	Der User kann die Einstellungen frei von den Dropdown-Felder eingeben.
Test ID: 3.1	
Name	Eingabedialoge: Einstellungen

Testvoraussetzung	
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User wählt Boolean und klickt auf «Select».
Erwartetes Resultat	Die Y-Axis und Type Felder für die Einstellungen werden blockiert.
Test ID: 4.0	
Name	Button: Select
Testvoraussetzung	Es wurde mindestens 1 und maximal 10 Objekte selektiert
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen. • User wählt Objekte aus und klickt den Button «Select».
Erwartetes Resultat	• Die ausgewählten Objektnamen werden links neben die Comboboxen für die Einstellungen platziert. • Die Comboboxen werden aktiviert. • Die anderen Steuerelemente werden aktiviert.
Test 4.1	
Name	Button: Select
Testvoraussetzung	Es wurde mindestens 1 und maximal 10 Objekte selektiert
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen User wählt keine Objekte aus und klickt den Button «Select».
Erwartetes Resultat	Es werden keine Objekte hinzugefügt und keine Steuerelemente aktiviert.
Test 4.2	
Name	Button: Select
Testvoraussetzung	Es wurde mindestens 1 und maximal 10 Objekte selektiert
Testablauf	• User klickt auf Trend Button. • Wechsel auf UI «Trend Creator». • Objektnamen werden geladen User wählt 11 Objekte aus und klickt den Button «Select».
Erwartetes Resultat	• Es werden der Reihe nach Objekte eingefügt. Es werden aber nur 10 Objekte eingefügt. • Die Comboboxen werden aktiviert • Die anderen Steuerelemente werden aktiviert.

Test der erstellten Funktionen

Test ID: 5.0	
Name	Button: Trend (Sheet Overview)
Testvoraussetzung	TestCase Sheet ist ausgewählt
Testablauf	User klickt auf den Button Trend
Erwartetes Resultat	• Das UI wird geladen. • Alle möglichen Objekte werden vom Test in die Listbox eingetragen. • Alle Steuerelemente sind gesperrt bis auf die Listbox und der Select Button.
Test ID: 5.1	
Name	Button: Trend (Sheet Overview)
Testvoraussetzung	TestCase Sheet ist ausgewählt
Testablauf	User klickt auf den Button Trend
Erwartetes Resultat	• Das UI wird geladen. • Alle möglichen Objekte werden vom Test in die Listbox eingetragen. • Alle Steuerelemente sind gesperrt bis auf die Listbox und der Select Button. • Die Objektnamen werden verändert: Kennung:AliasName.PropertyShortName - P Priority
Test ID: 5.2	
Name	Button: Trend (Sheet Overview)
Testvoraussetzung	TestCase Sheet ist ausgewählt
Testablauf	• Das UI wird geladen. • Es existiert bereits ein Trend Sheet mit diesem Namen.
Erwartetes Resultat	Die Einstellungen für die Diagrammerstellung des existierenden Sheets wird übernommen. Der User kann die Einstellungen bei Bedarf ändern.
Test ID: 5.2	
Name	Button: DelGenerated
Testvoraussetzung	Test ist selektiert
Testablauf	• User klickt auf den Button «DelGenerated».

	• Alles was zum Test gehört wird gelöscht.
Erwartetes Resultat	Neu auch die Trends und Tabellenblätter.
Test ID: 5.3	
Name	Button: Create Trend
Testvoraussetzung	• Objekte sind selektiert. • Alle Einstellungen sind gewählt.
Testablauf	User klickt auf den Button «Create Trend»
Erwartetes Resultat	Werte werden in eine Hilfstabelle geschrieben. Diagramm wird erstellt und auf einem zusätzlichen Tabellenblatt abgespeichert.
Test ID: 5.4	
Name	Button: Create Trend
Testvoraussetzung	Objekte sind ausgewählt
Testablauf	• Einstellungen wurden nicht vollständig vorgenommen. • User klickt auf den Button Create Trend.
Erwartetes Resultat	Für die fehlenden Angaben werden Standard Werte genommen
Test ID: 5.5	
Name	Button: Create Trend
Testvoraussetzung	Einstellungen wurden vorgenommen
Testablauf	Der User hat **keine** Objekte ausgewählt User klickt auf den Button Create Trend.
Erwartetes Resultat	Fehlermeldung an den User erscheint, dass er kein leeres Diagramm erstellen kann.
Test ID: 5.6	
Name	Button: Create Trend
Testvoraussetzung	Objekte sind gewählt
Testablauf	User nimmt folgende Einstellungen für das Objekt Analog1 vor: Range: Ignored LineType: Line Type: Step Y-Axis: Left
Erwartetes Resultat	Hilfstabelle wird folgendermassen aufgebaut: 5 Zeilen werden mit Testwert befüllt Bei einem Bereich wird der kleinere Wert genommen. Diagramm wird erstellt.

Test ID: 5.7	
Name	Button: Create Trend
Testvoraussetzung	TestCase Sheet ist ausgewählt Objekte sind gewählt.
Testablauf	User nimmt folgende Einstellungen für das Objekt Analog1 vor: **Range: Show** LineType: Line Type: Step Y-Axis: Left
Erwartetes Resultat	Hilfstabelle wird folgendermassen aufgebaut: Zusätzliche Spalte wird erstellt Maximalwerte von Bereich werden hineingeschrieben. Diagramm wird erstellt.
Test ID: 5.8	
Name	Button: Create Trend
Testvoraussetzung	TestCase Sheet ist ausgewählt Objekte sind gewählt
Testablauf	User nimmt folgende Einstellungen für das Objekt Analog1 vor: Range: Ignored **LineType: Pointed** Type: Step Y-Axis: Left
Erwartetes Resultat	Hilfstabelle wird folgendermassen aufgebaut: 5 Zeilen werden mit Testwert befüllt. Diagramm wird erstellt. Objekt wird mit gepunkteter Linie dargestellt.
Test ID: 5.9	
Name	Button: Create Trend
Testvoraussetzung	TestCase Sheet ist ausgewählt Objekte sind gewählt
Testablauf	User nimmt folgende Einstellungen für das Objekt Analog1 vor: Range: Ignored LineType: Line **Type: Continuous** Y-Axis: Left
Erwartetes Resultat	Hilfstabelle wird folgendermassen aufgebaut:

	Testwert wird interpoliert.
Test ID: 5.10	
Name	Button: Create Trend
Testvoraussetzung	TestCase Sheet ist ausgewählt Objekte sind gewählt.
Testablauf	User nimmt folgende Einstellungen für das Objekt Analog1 vor: Range: Ignored LineType: Line Type: Step **Y-Axis: Right**
Erwartetes Resultat	Hilfstabelle wird folgendermassen aufgebaut: 5 Zeilen werden mit Testwert befüllt. Diagramm wird erstellt, Objekt wird in der Sekundärachse angezeigt.

Useability Fragen:

1. Ist klar, wie man die Objekte aussuchen kann?
2. Ist klar, wie man Einstellungen zu ihnen vornimmt?
3. Ist klar, wie man die Überschrift des Diagrammes/den Namen des Sheets festlegt?
4. Ist die Idee und Umsetzung der verschiedenen Seiten bzw. Diagrammen klar?
5. Ist das Vorgehen verständlich?
6. Ist das UI übersichtlich und verständlich aufgebaut? Weiss man wo sich was befindet? Musstest du lange suchen bis du einzelne Elemente des UIs entdeckt hast?
7. Ist das UI angenehm zu betrachten oder sehr kompliziert?
8. Ist das Diagramm verständlich?
9. Hast du sonstige Hinweise, Anregungen oder Tipps?

2.3.2 Ergebnis

Akzeptanztest Ergebnis

Nr.	Test Ergebnis	Kommentar	Datum	Wer
1.0	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
1.1	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
1.2	Nein	Wird noch nicht abgefragt, deshalb Ergebnis «Nein»	28.03.2018	Patrick Schöpfer
1.3	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
2.0	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
3.0	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
3.1	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
4.0	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
4.1	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
4.2	Nein	Nicht alle Steuerelemente werden korrekt freigegeben. Bei Boolean werden alle freigegeben, dürfen aber nur 2.	28.03.2018	Patrick Schöpfer
5.0	Ja	Ergebnis wie erwartet	28.03.2018	Patrick Schöpfer
5.1	Nein	Objektnamen werden nicht verändert	29.03.2018	Patrick Schöpfer
5.2	Nein	Wurde noch nicht implementiert.	29.03.2018	Patrick Schöpfer
5.3	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer
5.4	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer
5.5	Nein	Fehlermeldung bisher nicht implementiert	29.03.2018	Patrick Schöpfer
5.6	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer
5.7	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer
5.8	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer
5.9	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer
5.10	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer

Useability Resultat

Bei den Useability Fragen haben Anton und ich uns unterhalten und er hat mir das Feedback gegeben, dass einige Felder, welche ich auch im Akzeptanztest gesehen habe noch nicht auf die Eingaben getestet wird. Ich habe die Fehler behoben. Weiterhin hatte Anton Kryenbuehl mir gesagt, dass er das Programm sonst verständlich und leicht zu bedienen findet. Damit wurde mein Ziel für die Useability erfüllt.

2.3.3 Naschtests

Nr.	Test Ergebnis	Kommentar	Datum	Wer	Ausbessern bis
1.0	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
1.1	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
1.2	Ja	Wurde implementiert. Reagiert nun wie erwartet	29.03.2018	Patrick Schöpfer	
1.3	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
2.0	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
3.0	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
3.1	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
4.0	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
4.1	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
4.2	Ja	Wurde verändert. Ich habe die Benennung zum abgeändert, um den Boolean zu erkennen. Booleans werden nun mit :Boolean im Namen angezeigt	29.03.2018	Patrick Schöpfer	
5.0	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.1	Nein	Objektnamen werden nicht verändert	29.03.2018	Patrick Schöpfer	19.04.2018
5.2	Ja	Wurde nun implementiert und reagiert wie erwartet	29.03.2018	Patrick Schöpfer	
5.3	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.4	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.5	Ja	Fehlermeldung wurde nun implementiert	29.03.2018	Patrick Schöpfer	

5.6	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.7	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.8	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.9	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	
5.10	Ja	Ergebnis wie erwartet	29.03.2018	Patrick Schöpfer	

3. Auswerten

Ich habe alle meine Tests ausgewertet und nach dem 1. Durchlauf gesehen, dass noch einiges zu verbessern ist. Ich habe nicht alle Eingabedialoge auf Falscheingaben geprüft, was zu abstürzen führte. Dies darf natürlich nicht passieren und ich habe die Fehler in der dafür vorgesehenen Fehlerverbesserung verbessert.

Im Nachtest konnte ich alles mit Ja, also korrekt, bewerten. Für mich bedeutet das, dass das Programm alles erfüllt, was es erfüllen muss. Es gibt jedoch kleinere Fehler, die den Betrieb nicht direkt stören, jedoch behoben werden sollten. Dies wird nach der IPA durchgeführt.

3.1 Schlussbericht

Für mich war die IPA eine anstrengende, jedoch auch eine erfolgreiche Zeit. Ich kannte davon einiges schon, wie z.B. das TsNet und habe aber auch viel zum ersten Mal gesehen, wie z.B. die Diagramm Funktion oder das Ganze abwickeln eines Projektes innerhalb von 2 Wochen. Am Anfang empfand ich dies als Herausforderung, ich gewöhnte mich jedoch schnell daran.

Ich habe IPERKA als Projektmanagement Methode ausgewählt, es half mir die Aufgaben gut einteilen zu können und ich konnte mir damit einen guten Überblick verschaffen.

Die beiden Phasen, Information und Planen, verliefen gut und nach Zeitplan. Ich hatte keine grossen Schwierigkeiten mit dem Zeitplan und konnte ihn rechtzeitig abgeben. Auch die Entscheidungsphase verlief ohne schwerwiegende Probleme.

In der Realisierungsphase habe ich mich leicht verschätzt. Ich habe zu wenig Zeit für die Implementierung vorgesehen und bin so aus dem Zeitplan geraten. Die Zeit konnte ich jedoch mit meinen geplanten Reserven und anderen, weniger lang dauernden, Tätigkeiten wieder aufholen. Ich werde in Zukunft mehr Zeit für Implementierungen einschätzen, da man bei allfälligen Fehler oft nicht genau einschätzen kann, wie lange man braucht um sie zu lösen.

Die Kontrolle verlief gut, auch wenn bei Anton Kryenbuehl aufgefallen ist, dass gewisse Funktionen nicht für die früheren Office Versionen verfügbar sind. Dies konnte ich beheben, jedoch noch nicht ausgiebig testen.

Ich bin am Schluss wieder im Zeitplan gewesen und habe ein funktionierendes Programm abgegeben. Es hat noch kleine Fehler, die ich aber noch beheben werde. Diese konnte ich nicht durch fehlendes Wissen nicht umsetzen, sondern war ein wenig unter Zeitdruck und musste Priorisieren, welche Funktionen zuerst umzusetzen waren. Ich werde dies nach der IPA jedoch nachholen.

Ich bin im Grossen und Ganzen zufrieden mit meiner Leistung.

4. Glossar

Begriff	Beschreibung
ABT	Das Automation Building Tool ist ein Gebäudeautomationstool für die Wartung der Objekte.
Akzeptanztest	Durch die Akzeptanztests wird herausgefunden, ob das Programm funktioniert, wie erwartet. Es wird überprüft ob alle Anforderungen und Kriterien erfüllt werden.
Columns	Eine Spalte in einem Tabellenblatt.
Controller	Der Controller ist ein Automationsgerät, welches man programmieren kann. Die Applikation wird durchs ABT auf den Controller geladen.
Flussdiagramm	Flussdiagramme zeigen den Ablauf des Programmes in einem Fluss. Es ist weniger detailliert als das Struktogramm. Es dient in diesem Dokument hauptsächlich zur Hilfe für spätere Entwickler.
Funktionen	Gruppe von Befehlen oder Anweisungen. Gibt Rückgabewerte zurück.
IPA	Die IPA wird am Ende der Lehre gemacht. Bei dieser Arbeit wird überprüft, ob der Kandidat die benötigten Kenntnisse besitzt um die Ausbildung abschliessen zu können.
IPERKA	IPERKA ist die Projektmanagement Methode. Sie wird verwendet, um das Projekt richtig planen zu können.
Makros	Sind Befehlsfolgen die von Office Programmen ausgeführt werden kann.
Modul	Ein Modul ist ein Container für Code.
Planungsdiagramm	Planungsdiagramme sind Diagramme, die der Planung helfen. Zum Beispiel Struktogramme oder Flussdiagramme.
Prozeduren	Gruppe von Befehlen oder Anweisungen. Gibt keine Rückgabewerte zurück.
Range	Ein bestimmter Bereich im Test angeben oder ein bestimmter Bereich auf einem Tabellenblatt.
Rows	Eine Zeile in einem Tabellenblatt.
Struktogramm	Struktogramme sind Nassi-Shneiderman-Diagramme. Sie stellen den Ablauf des Programmes dar.
Trend	
TsNet	Das TsNet ist ein Testautomationstool mit dem wir Controller auf bestimmte Bedingungen testen

Useability Test	Die Useability Tests sind Tests, welche die Brauchbarkeit des Programmes testet. Bei diesen Tests ist besonders, dass man der anderen Person keinerlei Anweisungen gibt, wie sie vorgehen muss. Sie versucht herauszufinden wie es funktioniert um zu sehen, ob es leicht verständlich umgesetzt wurde.
User Interface / UI	Das User Interface ist eine Schnittstelle, die vom User gesteuert wird.
VBA	VBA ist die Programmiersprache mit der man Office Programme bzw. Makros programmieren kann. Ausgeschrieben heisst es «Visual Basic for Applications»

Tabelle 14: Glossar

5. Quellen

Detaillierte Aufgabenstellung	www.pkOrg.ch
IPERKA	https://www.vssm.ch/sites/default/files/vssm/BB/docs/AusbildungEFZ/IPERKA_1_4_D2017.pdf http://beni.ch/downloads/lichtblicke/IPERKA%20Infos.pdf
Struktogramme	http://www.struktogrammer.ch/
Flussdiagramme	https://www.lucidchart.com/pages/de/was-ist-ein-flussdiagramm
Diagrammerstellung	https://msdn.microsoft.com/en-us/library/office/ff194426(v=office.14).aspx

Tabelle 15: Quellen

6. Anhang

Im Anhang befindet sich mein Code. Ich habe das gesamte Modul «mdl_Trend» eingefügt, da dies der Hauptteil meiner Arbeit war.

```
' ---------------------------------------------------------------------------------------------------'
' Name:          mdl_Trend                                                                           '
' File:          TsNetV2.1.3.3.xlsm                                                                  '
' ---------------------------------------------------------------------------------------------------'
' Copyright @2016,  SIEMENS Building Technologies                                                   '
' ---------------------------------------------------------------------------------------------------'
' OS:            Win64                                                                               '
' Lang:          Microsoft Visual Basic 6.5                                                          '
' ---------------------------------------------------------------------------------------------------'
' Project:       TsNet                                                                               '
' Author:        Patrick Schöpfer                                                                    '
' Version        V.2.1.3.3                                                                           '
' Date           04-04-2018                                                                          '
' ---------------------------------------------------------------------------------------------------'
' Description:                                                                                       '
'   Einzelne Funktionen zur Bearbeitung des Diagramms und der zugehörigen Hilfstabelle               '
' ---------------------------------------------------------------------------------------------------'
' History:       26-03-2018 Patrick Schöpfer                                                         '
'                Modul Creation for IPA                                                              '
'                LoadTrendData()                                                                     '
'                SelectObjects()                                                                     '
'                GetMaxValue()                                                                       '
'                GetMinValue()                                                                       '
'                GetFalseValue()                                                                     '
'                GetTrueValue()                                                                      '
'                GetNextBool()                                                                       '
'                BoolCheck()                                                                         '
'                TypeContinuous()                                                                    '
'                RangeShow()                                                                         '
'                TrendCreation()                                                                     '
'                CreateHelpTable()                                                                   '
'---------------------------------------------------------------------------------------------------'

'-----------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Dies ist die Hautpfunktion. Sie ruft alle anderen Funktionen auf und erstellt die Hilfstabelle                      '
'-----------------------------------------------------------------------------------------------------------------------------------'
Sub CreateHelpTable()
    Dim strDataArray(9, 7) As String
    Dim intColumnCounter As Integer
    Dim intColumnCounterTrend As Integer
    Dim intRowCounter As Integer
    Dim intTestStepCounter As Integer
    Dim rngTestStep As Range
    Dim strTestStepArray() As String
    Dim intDataCounter As Integer
    Dim strAxisArray(10) As String
    Dim strBoolArray(10) As String
    Dim strArrayFalse(10) As String
    Dim intMaxRows As Integer
    Dim intObjectCounter As Integer
    Dim strSplitArray() As String
    Dim strSheetName As String
    Dim strTestSheetName As String
    Dim BoolSecondTrend As Boolean
    Dim BoolThirdTrend As Boolean
```

```
Dim BoolFourthTrend As Boolean
Dim intTrendsToCreate As Integer
Dim strRangeIgnored As String
Dim strLineType As String
Dim strYAxis As String
Dim strLineTypeArray(10) As String
Dim strYAxisArray(10) As String
Dim strRangeArray(10) As Integer
Dim intRangeCounter As Integer
Dim intLineTypeCounter As Integer
Dim strTrendTitle(1 To 4) As String
Dim strTrendHeader(3) As String
Dim ws As Worksheet
Dim strTestNameSplitter() As String
Dim nextBoolValue As Double
Dim strDiagName As String
Dim intProcessCounter As Integer

If TrendCreator.Controls("lblObjectName10") = "" Then
   MsgBox "Choose at least one Test Step"
   Exit Sub
End If

'Die SheetNamen der Trends werden in ein Array gespeichert
strTrendHeader(intTrendsToCreate) = TrendCreator.Controls("TextBoxTrendTitle10")
intTrendsToCreate = 1
'Dies wird gesplittet da wir nur den reinen Testnamen benötigen
strTestNameSplitter() = Split(TrendCreator.Controls("lblSheetName10"), "_")
strTrendTitle(intTrendsToCreate) = "Trend" & strTestNameSplitter(2) & "_" & intTrendsToCreate

'Überprüft, wie viele Diagramme zu erstellen sind und weist die Namen, welche sie haben sollen dem Array zu
If TrendCreator.Controls("lblObjectName20") = "" Then
Else
   strTrendHeader(intTrendsToCreate) = TrendCreator.Controls("TextBoxTrendTitle20")
   intTrendsToCreate = intTrendsToCreate + 1
   strTrendTitle(intTrendsToCreate) = "Trend" & strTestNameSplitter(2) & "_" & intTrendsToCreate
End If

If TrendCreator.Controls("lblObjectName30") = "" Then
Else
   strTrendHeader(intTrendsToCreate) = TrendCreator.Controls("TextBoxTrendTitle30")
   intTrendsToCreate = intTrendsToCreate + 1
   strTrendTitle(intTrendsToCreate) = "Trend" & strTestNameSplitter(2) & "_" & intTrendsToCreate
End If

If TrendCreator.Controls("lblObjectName40") = "" Then
Else
   strTrendHeader(intTrendsToCreate) = TrendCreator.Controls("TextBoxTrendTitle40")
   intTrendsToCreate = intTrendsToCreate + 1
   strTrendTitle(intTrendsToCreate) = "Trend" & strTestNameSplitter(2) & "_" & intTrendsToCreate
End If



'Der ganze Code wird für die Anzahl der zu erstellenden Diagrammen wiederholt
While intTrendsToCreate > 0
   strTestNameSplitter() = Split(TrendCreator.Controls("lblSheetName10"), "_")
   strDiagName = strTestNameSplitter(2)
   Erase strDataArray
   intColumnCounter = 2
   intColumnCounterTrend = 1
```

```
    intObjectCounter = 0
    'Der Wert der Comboboxen wird in das strDataArray geschrieben
    For i = 0 To 9
        If TrendCreator.Controls("lblObjectName" & intTrendsToCreate & i) = "" Then
        Else
            strDataArray(i, 4) = TrendCreator.Controls("lblObjectName" & intTrendsToCreate & i)
            strDataArray(i, 3) = TrendCreator.Controls("ComboBoxYAxis" & intTrendsToCreate & i)
            strYAxis = strDataArray(i, 3)
            strDataArray(i, 0) = TrendCreator.Controls("ComboBoxLineType" & intTrendsToCreate & i)
            strLineType = strDataArray(i, 0)
            strDataArray(i, 1) = TrendCreator.Controls("ComboBoxRange" & intTrendsToCreate & i)
            strRangeIgnored = strDataArray(i, 1)
            strDataArray(i, 2) = TrendCreator.Controls("ComboBoxType" & intTrendsToCreate & i)
            If IsNumeric(TrendCreator.Controls("TextBoxFrom" & intTrendsToCreate & "0")) = True And
IsNumeric(TrendCreator.Controls("TextBoxTo" & intTrendsToCreate & "0")) = True Then
                strDataArray(i, 5) = TrendCreator.Controls("TextBoxFrom" & intTrendsToCreate & "0")
                strDataArray(i, 6) = TrendCreator.Controls("TextBoxTo" & intTrendsToCreate & "0")
            Else
                MsgBox "TestSteps has to be Numeric"
                Exit Sub
            End If
            strDataArray(i, 7) = TrendCreator.Controls("lblSheetName10")
            intObjectCounter = intObjectCounter + 1
            If strDataArray(i, 1) = "Show" And strDataArray(i, 2) = "Continuous" Then
                MsgBox "You can't use Continuous and Show at the same time."
                Exit Sub
            End If
        End If
    Next


    'Die Hilfstabelle wird erstellt und benannt
    Worksheets("Trendtable_Template").Copy After:=Worksheets("Choices")
    strSheetName = strTrendTitle(intTrendsToCreate)
    'Falls ein Sheet mit demselben Namen existiert, wird gefragt ob man das Sheet überschreiben möchte
    For Each ws In ActiveWorkbook.Worksheets
        If ws.Name = strTrendTitle(intTrendsToCreate) Then
            If MsgBox("Press Yes to overwrite the existing Trend Sheet", vbYesNo) = vbYes Then
                On Error Resume Next
                If intProcessCounter = 0 Then
                    Sheets(strDiagName & "_Diagram").Delete
                End If
                ws.Delete
            Else
                Exit Sub
            End If
        End If
    Next

    'Schreiben der Daten in die Hilfstabelle
    ActiveSheet.Name = strSheetName
    For i = 0 To 9
        intRowCounter = 2
        For X = 0 To 4 'dynamisch mit Konstanten TODO
            'Trägt alles was sich im zweidimensionale Array befindet (Objektnamen, Einstellungen, TestSheet und die Von Bis
TestSteps) in die Hilfstabelle ein
            Cells(intRowCounter, intColumnCounter).Value = strDataArray(i, X)
            intRowCounter = intRowCounter + 1
        Next
        intColumnCounter = intColumnCounter + 1
    Next
```

```
    [ValueFromTestStep] = strDataArray(0, 5)
    [ValueToTestStep] = strDataArray(0, 6)
    [TestSheetName] = strDataArray(0, 7)
    strTestSheetName = strDataArray(0, 7)

    'Anzahl der ausgewählten TestSteps
    If strDataArray(0, 6) > strDataArray(0, 5) Then
        intTestStepCounter = strDataArray(0, 6) - strDataArray(0, 5) + 1
    Else
        'Wenn der User zuerst eine höhere Zahl eingegeben hat
        MsgBox "Please enter the lower Value in the first TextBox for Teststeps"
        Exit Sub
    End If
    'Holt zuerst die Teststep namen aus dem TestStep Sheet
    ReDim strTestStepArray(intTestStepCounter - 1)
    'Holt dann die Daten zu den TestSteps
    intColumnCounter = 7

    'Daten von Test werden ausgelesen
    For i = 0 To 9
        intRowCounter = 26
        intDataCounter = 0
        While intTestStepCounter > intDataCounter
            'Überprüfung ob Objekt exitsiert
            If Sheets(strTestSheetName).Cells(10, intColumnCounter) = strDataArray(i, 4) Then
                'Objekte werden in Array gespeichert
                strTestStepArray(intDataCounter) = Sheets(strTestSheetName).Cells(intRowCounter, intColumnCounter).Value
                intRowCounter = intRowCounter + 1
                intDataCounter = intDataCounter + 1
                'Wenn Objekte nicht gefunden wurde und die Suche abgeschlossen ist, wird eine fehlermeldung ausgegeben
            ElseIf Sheets(strTestSheetName).Cells(9, intColumnCounter).Value = "End" Then
                MsgBox "Test can't be found"
                TrendCreator.Hide
                Exit Sub
            Else
                intColumnCounter = intColumnCounter + 1
            End If
        Wend


        intRowCounter = 6
        intDataCounter = 0
        intColumnCounter = 2

        'Fügt die Daten vom Test in die Hilfstabelle
        While intDataCounter < intTestStepCounter
            'Der Maximale Wert von X ist wie oft ein Test eingetragen wird. Also 5 mal
            For X = 0 To 4
                'Überprüft ob es sich bei den Objekten um Bereiche handelt oder ob sie ignoriert werden können
                If Sheets(strSheetName).Cells(3, intColumnCounter + i) = "Ignore" Or Sheets(strSheetName).Cells(3,
intColumnCounter + i) = "" Then
                    'Wenn sie ignoriert(standard bei keiner angabe) werden können, werden die Bereiche in normale Zahlen umgeformt
                    If InStr(strTestStepArray(intDataCounter), "..") > 0 Then
                        strSplitArray() = Split(strTestStepArray(intDataCounter), "..")
                        strTestStepArray(intDataCounter) = strSplitArray(0)
                    End If
                End If
                Sheets(strSheetName).Cells(intRowCounter + 1, intColumnCounterTrend + 1).Value =
strTestStepArray(intDataCounter)
                intRowCounter = intRowCounter + 1
                'Schreibt für jeden 3. Wert die Testnamen in die erste Spalte der Hilfstabelle
```

```
            If X = 2 And intColumnCounterTrend = 1 Then
                Sheets(strSheetName).Cells(intRowCounter, intColumnCounterTrend).Value = "test" & intDataCounter
            End If
        Next
        intDataCounter = intDataCounter + 1
    Wend
    intColumnCounterTrend = intColumnCounterTrend + 1
Next

intMaxRows = intTestStepCounter * 5
intAmountOfBool = 0
intMinValue = 1000

'Folgend werden die Daten den Einstellungen entsprechend abgeändert.
While Sheets(strSheetName).Cells(6 + 1, intColumnCounter) <> ""
    Set rngData = Sheets(strSheetName).Range(Cells(6 + 1, intColumnCounter), Cells(intMaxRows + 6, intColumnCounter))
    If Sheets(strSheetName).Cells(4, intColumnCounter) = "Continuous" Then
        'Wenn Continuous in den Einstellungen steht, wird die Funktion für die Interpolierung aufgerufen und die Aktuelle Spalte
mitgegeben
        Call TypeContinuous(rngData, intTestStepCounter)
    End If
    If Sheets(strSheetName).Cells(3, intColumnCounter) = "Show" Then
        'Wenn Show in den Einstellungen steht, wird die Funktion für die Anzeige des Bereiches aufgerufen und die Aktuelle
Spalte mitgegeben
        Call RangeShow(rngData, strSheetName)
    End If
    If Sheets(strSheetName).Cells(5, intColumnCounter) = "Right" Then
        'strAxisArray(yAxisCounter) = Sheets(strSheetName).Cells(6, intColumnCounter) ToDOKommEntarLöschen
        'Der Objektname wird abgeändert und ein Pfeil eingefügt, damit man sehen kann, welches Objekt als sekundärachse
angezeigt wird
        Sheets(strSheetName).Cells(6, intColumnCounter) = Sheets(strSheetName).Cells(6, intColumnCounter) & "-->"
       ' yAxisCounter = yAxisCounter + 1 ToDOKommEntarLöschen
    End If
    If Sheets(strSheetName).Cells(6 + 1, intColumnCounter) = "" Then
        'Wenn die Minimal und Maximal Werte = 0 sind werden sie angepasst und ein Standard Wert wird zugewiesen
        If intMinValue = 0 And intMaxValue = 0 Then
            intMinValue = 0
            intMaxValue = 100
        End If
    End If
    If BoolCheck(rngData, strSheetName, intTrendsToCreate) = True Then
        'Wenn das Objekt ein Boolean ist, wird es erstmal in ein Array gespeichert
        strBoolArray(intAmountOfBool) = Sheets(strSheetName).Cells(6, intColumnCounter)
        Sheets(strSheetName).Cells(6, intColumnCounter) = strBoolArray(intAmountOfBool) & ":Boolean"
        strBoolArray(intAmountOfBool) = Sheets(strSheetName).Cells(6, intColumnCounter)
        intAmountOfBool = intAmountOfBool + 1
    Else
        'Wenn es kein Boolean ist, wird der Maximal und Minimal Wert für möglicherweise Vorhandene Booleans ausgerechnet
        If intMaxValue < GetMaxValue(rngData, strSheetName, intTrendsToCreate) Then
            intMaxValue = GetMaxValue(rngData, strSheetName, intTrendsToCreate)
        End If
        If intMinValue > GetMinValue(rngData, strSheetName, intTrendsToCreate) Then
            intMinValue = GetMinValue(rngData, strSheetName, intTrendsToCreate)
        End If
    End If
    intColumnCounter = intColumnCounter + 1
Wend

'Die Bearbeitung der Boolean Werte beginnt hier
For n = 0 To 10
    If strBoolArray(n) <> "" Then
```

```
            intColumnCounter = 2
            intRowCounter = 7
            'überprüft ob die aktuelle ObjektNamen Zeile im TrendSheet leer ist
            While Sheets(strSheetName).Cells(intRowCounter - 1, intColumnCounter) <> ""
               'überprüfen ob der die aktuelle Zelle dem Bool entspricht
               If Sheets(strSheetName).Cells(intRowCounter - 1, intColumnCounter) = strBoolArray(n) Then
                  While Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) <> ""
                     If n > 0 Then
                        nextBoolValue = GetNextBool(intMinValue, intMaxValue, n) 'vlt + 1
                        If Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = 0 Then
                           'Rufe die funktion auf um den False Wert des Bools zu bekommen
                           Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = GetFalseValue(intMinValue, intMaxValue,
intAmountOfBool, n + 1, 0) + nextBoolValue
                           strArrayFalse(n) = Sheets(strSheetName).Cells(intRowCounter, intColumnCounter)
                        ElseIf Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = 1 Then
                           'Rufe die funktion auf um den True Wert des Bools zu bekommen
                           Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = GetTrueValue(intMinValue, intMaxValue,
intAmountOfBool, n + 1, 1) + nextBoolValue
                        End If

                     Else
                        If Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = 0 Then
                           'Rufe die funktion auf um den False Wert des Bools zu bekommen
                           Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = GetFalseValue(intMinValue, intMaxValue,
intAmountOfBool, n + 1, 0)
                           strArrayFalse(n) = Sheets(strSheetName).Cells(intRowCounter, intColumnCounter)
                        ElseIf Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = 1 Then
                           'Rufe die funktion auf um den True Wert des Bools zu bekommen
                           Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = GetTrueValue(intMinValue, intMaxValue,
intAmountOfBool, n + 1, 1)
                        End If
                     End If
                     intRowCounter = intRowCounter + 1
                  Wend
               Else
                  intColumnCounter = intColumnCounter + 1
               End If
            Wend
         End If
      Next

      intRowCounter = 6
      intColumnCounter = 12
      'Die False Werte werden in eine zusätzliche Spalte gespeichert
      For n = 0 To 10
         'Es wird inkrementiert bis man bei Spalte 12 ist
         While Cells(intRowCounter, intColumnCounter) <> ""
            intColumnCounter = intColumnCounter + 1
         Wend
         'Die Werte für die False Linie werden eingetragen
         If strArrayFalse(n) <> "" Then
            Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = "False"
            Sheets(strSheetName).Range(Cells(intRowCounter + 1, intColumnCounter), Cells(intRowCounter + intTestStepCounter *
5, intColumnCounter)) = strArrayFalse(n)
            intColumnCounter = intColumnCounter + 1
         End If
         strArrayFalse(n) = ""
      Next
      'Die Hilfstabelle wird geschützt
      Sheets(strSheetName).Protect
```

```
        intRowCounter = 6
        intColumnCounter = 2
        intProcessCounter = intProcessCounter + 1

        'Diagramm Erstellung wird hier aufgerufen
        Call TrendCreation(intObjectCounter, ActiveSheet.Name, intTestStepCounter, intTrendsToCreate,
strTrendHeader(intTrendsToCreate - 1), strDiagName, intProcessCounter)
        intTrendsToCreate = intTrendsToCreate - 1
    Wend
End Sub

'-----------------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Diese Funktion überprüft ob schon ein TrendSheet mit diesem Namen existiert. Falls ja, werden die Einstellungen
übernommen'
'-----------------------------------------------------------------------------------------------------------------------------------------------'
Sub LoadTrendData(strTrendSheet)
    Dim intPageNumber As Integer
    Dim intColumnCounter As Integer
    Dim intRowCounter As Integer
    Dim ObjectNameSplitter() As String
    Dim intMaxObjectCounter As Integer

    For Each ws In Worksheets
        For intPageNumber = 1 To 4
            intColumnCounter = 2
                If ws.Name = "Trend" & strTrendSheet & "_" & intPageNumber Then
                    'Die Einstellungen werden pro Seite für jedes Steuerelement zugeordnet, falls ein solches Sheet existiert
                    intMaxObjectCounter = TrendCreator.Controls("ListBoxObjects" & intPageNumber & 0).ListCount
                    For intSettingRow = 0 To 9
                        intRowCounter = 2
                        TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intSettingRow) =
Sheets(ws.Name).Cells(intRowCounter, intColumnCounter)
                        TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intSettingRow).Enabled = True
                        intRowCounter = intRowCounter + 1
                        TrendCreator.Controls("ComboBoxRange" & intPageNumber & intSettingRow) =
Sheets(ws.Name).Cells(intRowCounter, intColumnCounter)
                        TrendCreator.Controls("ComboBoxRange" & intPageNumber & intSettingRow).Enabled = True
                        intRowCounter = intRowCounter + 1
                        TrendCreator.Controls("ComboBoxType" & intPageNumber & intSettingRow) =
Sheets(ws.Name).Cells(intRowCounter, intColumnCounter)
                        TrendCreator.Controls("ComboBoxType" & intPageNumber & intSettingRow).Enabled = True
                        intRowCounter = intRowCounter + 1
                        TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intSettingRow) =
Sheets(ws.Name).Cells(intRowCounter, intColumnCounter)
                        TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intSettingRow).Enabled = True
                        intRowCounter = intRowCounter + 1
                        If intColumnCounter > 11 Then
                        Else
                            If InStr(Sheets(ws.Name).Cells(intRowCounter, intColumnCounter), "-->") > 0 Then
                                ObjectNameSplitter() = Split(Sheets(ws.Name).Cells(intRowCounter, intColumnCounter), "-->")
                                TrendCreator.Controls("lblObjectName" & intPageNumber & intSettingRow) = ObjectNameSplitter(0)
                            ElseIf InStr(Sheets(ws.Name).Cells(intRowCounter, intColumnCounter), ":Boolean") > 0 Then
                                ObjectNameSplitter() = Split(Sheets(ws.Name).Cells(intRowCounter, intColumnCounter), ":Boolean")
                                TrendCreator.Controls("lblObjectName" & intPageNumber & intSettingRow) = ObjectNameSplitter(0)
                                TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intSettingRow).Enabled = False
                                TrendCreator.Controls("ComboBoxType" & intPageNumber & intSettingRow).Enabled = False
                            Else
                                TrendCreator.Controls("lblObjectName" & intPageNumber & intSettingRow) =
Sheets(ws.Name).Cells(intRowCounter, intColumnCounter)
                                For i = 0 To intMaxObjectCounter - 1
```

```
                    If TrendCreator.Controls("ListBoxObjects" & intPageNumber & 0).List(i) =
Sheets(ws.Name).Cells(intRowCounter, intColumnCounter) Then
                      TrendCreator.Controls("ListBoxObjects" & intPageNumber & 0).Selected(i) = True
                    End If
                  Next
                End If
              End If
              intColumnCounter = intColumnCounter + 1
              If intSettingRow = 0 Then
                'Hier werden die Steuerelemente friegegeben.
                TrendCreator.Controls("TextBoxTrendTitle" & intPageNumber & intSettingRow).Enabled = True
                TrendCreator.Controls("TextBoxFrom" & intPageNumber & intSettingRow).Enabled = True
                TrendCreator.Controls("TextBoxFrom" & intPageNumber & intSettingRow) = Sheets("Trend" & strTrendSheet &
"_" & intPageNumber).[ValueFromTestStep]
                TrendCreator.Controls("TextBoxTo" & intPageNumber & intSettingRow).Enabled = True
                TrendCreator.Controls("TextBoxTo" & intPageNumber & intSettingRow) = Sheets("Trend" & strTrendSheet & "_" &
intPageNumber).[ValueToTestStep]
              End If
            Next
          End If
        Next
      Next
End Sub
'-----------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Diese Funktion überprüft die selektierten, fügt sie links neben die Einstellungen ein und gibt Steuerelemente frei.
'
'-----------------------------------------------------------------------------------------------------------------------------------'
Sub SelectObjects(intPageNumber)
  Dim intMaxCountObjects As Integer
  Dim intObjectCounter As Integer
  Dim BoolValue As Boolean
  Dim SheetName As String
  Dim intColumCounter As Integer
  Dim intRowCounter As Integer
  Dim intObjectNameCounter As Integer
  Dim intlblCounter As Integer
  Dim intMaxTestSteps As Integer
  Dim BoolDeselect As Boolean
  Dim intDeselectCounter As Integer

  intObjectNameCounter = 10
  intColumnCounter = 7

  SheetName = TrendCreator.lblSheetName10

  intMaxCountObjects = TrendCreator.Controls("ListBoxObjects" & intPageNumber & 0).ListCount
  'Mit select werden die anderen steuerelemente freigeschaltet
  TrendCreator.Controls("TextBoxTrendTitle" & intPageNumber & 0).Enabled = True
  TrendCreator.Controls("TextBoxFrom" & intPageNumber & 0).Enabled = True
  TrendCreator.Controls("TextBoxTo" & intPageNumber & 0).Enabled = True
  'Standard Werte werden eingefügt, dass nichts leer stehen kann
  TrendCreator.Controls("TextBoxFrom" & intPageNumber & 0).Value = 1

  'Es werden die Maximalen TestSteps ausgelesen und standardmässig eingetragen.
  intMaxTestSteps = Sheets(SheetName).Cells(Rows.count, 1).End(xlUp).Row
  intMaxTestSteps = intMaxTestSteps - Sheets(SheetName).[_Step].Row
  TrendCreator.Controls("TextBoxTo" & intPageNumber & 0).Value = intMaxTestSteps

  'Für jedes Objekt in der Liste wird überprüft, ob es selektiert ist
  For i = 0 To intMaxCountObjects - 1
    intDeselectCounter = 0
```

```
        BoolValue = True
        BoolDeselect = True
        intRowCounter = 26
        If intObjectCounter = 10 Then
            i = intMaxCountObjects - 1
        Else
            If TrendCreator.Controls("ListBoxObjects" & intPageNumber & 0).Selected(i) = True Then
                BoolDeselect = False
                'Falls eines selektiert ist, wird es eingetragen und überprüft ob es sich um einen Boolean handelt
                TrendCreator.Controls("lblObjectName" & intPageNumber & intObjectCounter) = TrendCreator.Controls("ListBoxObjects"
& intPageNumber & 0).List(i)
                If BoolValue = True Then
                    While Sheets(SheetName).Cells(intObjectNameCounter, intColumnCounter) <>
TrendCreator.Controls("ListBoxObjects" & intPageNumber & 0).List(i)
                        intColumnCounter = intColumnCounter + 1
                    Wend
                    While Sheets(SheetName).Cells(intRowCounter, intColumnCounter) <> ""
                        If Sheets(SheetName).Cells(intRowCounter, intColumnCounter) = "1" And BoolValue = True Or
Sheets(SheetName).Cells(intRowCounter, intColumnCounter) = "0" And BoolValue = True Then
                            BoolValue = True
                            intRowCounter = intRowCounter + 1
                        Else
                            BoolValue = False
                            intRowCounter = intRowCounter + 1
                        End If
                    Wend
                    For intDeselectCounter = intObjectCounter + 1 To 9
                        If TrendCreator.Controls("lblObjectName" & intPageNumber & intDeselectCounter) =
TrendCreator.ListBoxObjects10.List(i) Then
                            TrendCreator.Controls("lblObjectName" & intPageNumber & intDeselectCounter) = ""
                            TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intDeselectCounter) = ""
                            TrendCreator.Controls("ComboBoxRange" & intPageNumber & intDeselectCounter) = ""
                            TrendCreator.Controls("ComboBoxType" & intPageNumber & intDeselectCounter) = ""
                            TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intDeselectCounter) = ""
                            TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intDeselectCounter).Enabled = False
                            TrendCreator.Controls("ComboBoxRange" & intPageNumber & intDeselectCounter).Enabled = False
                            TrendCreator.Controls("ComboBoxType" & intPageNumber & intDeselectCounter).Enabled = False
                            TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intDeselectCounter).Enabled = False
                            BoolDeselect = True
                        End If
                    Next
                Else
                End If
                'intObjectCounter = intObjectCounter + 1
            'Falls es nicht selektiert ist, soll es aus dem UI herausgelöscht werden falls es drinn ist.
            Else
                For intDeselectCounter = 0 To 9
                If TrendCreator.Controls("lblObjectName" & intPageNumber & intDeselectCounter) =
TrendCreator.ListBoxObjects10.List(i) Then
                    TrendCreator.Controls("lblObjectName" & intPageNumber & intDeselectCounter) = ""
                    TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intDeselectCounter) = ""
                    TrendCreator.Controls("ComboBoxRange" & intPageNumber & intDeselectCounter) = ""
                    TrendCreator.Controls("ComboBoxType" & intPageNumber & intDeselectCounter) = ""
                    TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intDeselectCounter) = ""
                    TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intDeselectCounter).Enabled = False
                    TrendCreator.Controls("ComboBoxRange" & intPageNumber & intDeselectCounter).Enabled = False
                    TrendCreator.Controls("ComboBoxType" & intPageNumber & intDeselectCounter).Enabled = False
                    TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intDeselectCounter).Enabled = False
                    BoolDeselect = True
                End If
                Next
```

```
            End If
        End If
        'Gewisse Steuerelemente werden für Booleans gesperrt
        If BoolValue = True And intRowCounter > 26 Then
            TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intObjectCounter).Enabled = False
            TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intObjectCounter) = "Left"
            TrendCreator.Controls("ComboBoxType" & intPageNumber & intObjectCounter).Enabled = False
            TrendCreator.Controls("ComboBoxType" & intPageNumber & intObjectCounter) = "Step"
            TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intObjectCounter).Enabled = True
            TrendCreator.Controls("ComboBoxRange" & intPageNumber & intObjectCounter).Enabled = True
            intRowCounter = 26
        ElseIf intRowCounter > 26 Then
            On Error Resume Next
            TrendCreator.Controls("ComboBoxType" & intPageNumber & intObjectCounter).Enabled = True
            TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & intObjectCounter).Enabled = True
            TrendCreator.Controls("ComboBoxLineType" & intPageNumber & intObjectCounter).Enabled = True
            TrendCreator.Controls("ComboBoxRange" & intPageNumber & intObjectCounter).Enabled = True
        End If
        If BoolDeselect = False Then
            intObjectCounter = intObjectCounter + 1
        End If
    Next
        'Für jedes nicht verwendete Objekt werden die Steuerelemente gesperrt
        For i = 0 To 9
            If TrendCreator.Controls("lblObjectName" & intPageNumber & i) = "" Then
                TrendCreator.Controls("ComboBoxYAxis" & intPageNumber & i).Enabled = False
                TrendCreator.Controls("ComboBoxLineType" & intPageNumber & i).Enabled = False
                TrendCreator.Controls("ComboBoxRange" & intPageNumber & i).Enabled = False
                TrendCreator.Controls("ComboBoxType" & intPageNumber & i).Enabled = False
                TrendCreator.Controls("lblObjectName" & intPageNumber & i) = ""
            End If
        Next
End Sub

'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt den grössten aller Analogen Werte und man verwendet diesen für die Bool Werte berechnung.
'
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function GetMaxValue(rngData, strSheetName, intTrendsToCreate) As Integer
    Dim DataCell As Range
    Dim SplitArray() As String

    'Es wird überprüft ob es sich um einen Boolean handelt.
    If BoolCheck(rngData, strSheetName, intTrendsToCreate) = False Then
        For Each DataCell In rngData
            If InStr(DataCell, "..") > 0 Then
                SplitArray() = Split(DataCell, "..")
                'Wenn der aktuelle Zellenwert grösser als der Wert in der MaximalWert Variable ist, dann überschreibe den Wert auf der
Variable
                If SplitArray(1) > GetMaxValue Then
                    GetMaxValue = SplitArray(1)
                End If
            Else
                If DataCell > GetMaxValue Then
                    GetMaxValue = DataCell
                End If
            End If
        Next
    End If
End Function
```

```
'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt den kleinsten aller Analogen Werte und man verwendet diesen für die Bool Werte berechnung.
'
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function GetMinValue(rngData, strSheetName, intTrendsToCreate) As Integer
   Dim DataCell As Range
   Dim intMaxValue As Integer
   Dim SplitArray() As String

   GetMinValue = 1000

   'Es wird überprüft ob es sich um einen Boolean handelt.
   If BoolCheck(rngData, strSheetName, intTrendsToCreate) = False Then
      For Each DataCell In rngData
         If InStr(DataCell, "..") > 0 Then
            SplitArray() = Split(DataCell, "..")
            'Wenn der aktuelle Zellenwert grösser als der Wert in der MaximalWert Variable ist, dann überschreibe den Wert auf der
Variable
            If DataCell < GetMinValue Then
               GetMinValue = SplitArray(1)
            End If
         Else
            If DataCell < GetMinValue Then
               GetMinValue = DataCell
            End If
         End If
      Next
   End If
End Function

'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt den False Wert eines Booleans. Dieser muss speziell errechnet werden um gleichzeitig in einem
Diagramm mit Analogen Werten Dargestellt werden zu können                                                                                '
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function GetFalseValue(intMinValue, intMaxValue, intAmountOfBool, n, intBoolValue) As Double
   Dim intDifference As Integer
   'Berechnung der False Spalte
   intDifference = intMaxValue - intMinValue
   GetFalseValue = intMinValue - intAmountOfBool * intDifference / 5
End Function

'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt den True Wert eines Booleans. Dieser muss speziell errechnet werden um gleichzeitig in einem Diagramm
'mit Analogen Werten Dargestellt werden zu können                                                                                         '
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function GetTrueValue(intMinValue, intMaxValue, intAmountOfBool, n, intBoolValue) As Double
   Dim intDifference As Integer
   'Berechnung der True Werte des neuen Booleans
   intDifference = intMaxValue - intMinValue
   GetTrueValue = intBoolValue * intDifference / 10 + GetFalseValue(intMinValue, intMaxValue, intAmountOfBool, n, intBoolValue)
End Function
```

```
'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt den nächsten Wert, welcher der Falsch Wert des nächsten Booleans ist. Dieser muss speziell errechnet
‘Dieser muss speziell errechnet werden um 2 Booleans in einem Diagramm mit Analogen Werten gleichzeitig darstellen zu können
'
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function GetNextBool(intMinValue, intMaxValue, n) As Double
   Dim intDifference As Integer
   'Berechnung der Position, des nächsten Booleans
   intDifference = intMaxValue - intMinValue

   GetNextBool = 2 * (n) * intDifference / 10
End Function

'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Diese Funktion überprüft, ob es sich bei einer bestimmten Spalte um einen boolean handelt.
‘Es schickt ein True oder False zurück'
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function BoolCheck(rngData, strSheetName, intTrendsToCreate) As Boolean
   Dim ValueCell As Range

   BoolCheck = True
   For Each ValueCell In rngData
      'Es wird überprüft, ob sich nur 0 und 1 in den Daten befinden und ob es im UI korrekt gesperrt wurde
      If ValueCell = 0 And BoolCheck = True Or ValueCell = 1 And BoolCheck = True Or ValueCell < 0 And BoolCheck = True Then
         For i = 0 To 9
            If Sheets(strSheetName).Cells(ValueCell.Row - 1, ValueCell.Column) = TrendCreator.Controls("lblObjectName" &
intTrendsToCreate & i) Then
            ElseIf Sheets(strSheetName).Cells(ValueCell.Row - 1, ValueCell.Column) & ":Boolean" =
TrendCreator.Controls("lblObjectName" & intTrendsToCreate & i) Then
               If TrendCreator.Controls("ComboBoxYAxis1" & i).Enabled = False Then
                  BoolCheck = True
               Else
                  BoolCheck = False
               End If
            End If
         Next
      Else
         BoolCheck = False
      End If
   Next
End Function
```

```
'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt die Interpolierten Werte und trägt diese ein.                                                                    '
'-----------------------------------------------------------------------------------------------------------------------------------------'
Function TypeContinuous(rngData, intTests) As String
   'ReDim TypeContinuous(intTests * 5) As String
   Dim objCell As Range
   Dim intTestCounter As Integer
   Dim lngTestPair(4) As Double
   Dim intChangeProcedure As Integer
   Dim strMinusTest As String

   'intRowCounter = objCell.Row

   For Each objCell In rngData
      lngTestPair(intTestCounter) = objCell
      intTestCounter = intTestCounter + 1
      'Wir teilen die Tests in 5er Bündel auf und berechnen so die fehlenden Werte
      If intTestCounter = 5 Then
         If intChangeProcedure = 0 Then
            lngTestPair(0) = lngTestPair(0)
            lngTestPair(1) = lngTestPair(1)
            'Formelangaben
            lngTestPair(3) = lngTestPair(3) + (Cells(objCell.Row + 3, objCell.Column) - lngTestPair(3)) * 0.2
            lngTestPair(4) = lngTestPair(4) + (Cells(objCell.Row + 3, objCell.Column) - lngTestPair(4)) * 0.4
         Else
            ' überprüfe ob sich ein DoppelMinus hinter der gleichung verbirgt und das Programm beeinträchtigen könnten
            'Der Wert in der Zeile 1 berechnen
            strMinusTest = ((Cells(objCell.Row - 7, objCell.Column)) - lngTestPair(0)) * 0.4
            If InStr(strMinusTest, "-") > 0 Then
               lngTestPair(0) = lngTestPair(0) + (Cells(objCell.Row - 7, objCell.Column) - lngTestPair(0)) * 0.4
            Else
               lngTestPair(0) = lngTestPair(0) + ((Cells(objCell.Row - 7, objCell.Column)) - lngTestPair(0)) * 0.4
            End If

            'Den Wert in der Zeile 2 Berechnen
            strMinusTest = ((Cells(objCell.Row - 7, objCell.Column)) - lngTestPair(1)) * 0.2
            If InStr(strMinusTest, "-") > 0 Then
               lngTestPair(1) = lngTestPair(1) + (Cells(objCell.Row - 7, objCell.Column) - lngTestPair(1)) * 0.2
            Else
               lngTestPair(1) = lngTestPair(1) + ((Cells(objCell.Row - 7, objCell.Column)) - lngTestPair(1)) * 0.2
            End If

            'Den Wert in der Zeile 3 Berechnen
            strMinusTest = "+" & lngTestPair(3) + ((Cells(objCell.Row + 3, objCell.Column)) - lngTestPair(3)) * 0.2
            If InStr(strMinusTest, "+-") > 0 Then
               lngTestPair(3) = (-(lngTestPair(3) + (Cells(objCell.Row + 3, objCell.Column)) - lngTestPair(3)) * 0.2)
            Else
               lngTestPair(3) = lngTestPair(3) + ((Cells(objCell.Row + 3, objCell.Column)) - lngTestPair(3)) * 0.2
            End If

            'Den Wert in der Zeile 3 Berechnen
            strMinusTest = "+" & lngTestPair(3) + ((Cells(objCell.Row + 3, objCell.Column)) - lngTestPair(4)) * 0.2
            If InStr(strMinusTest, "+-") > 0 Then
               lngTestPair(4) = (-(lngTestPair(3) + (Cells(objCell.Row + 3, objCell.Column)) - lngTestPair(4)) * 0.2)
            Else
               lngTestPair(4) = lngTestPair(3) + ((Cells(objCell.Row + 3, objCell.Column)) - lngTestPair(4)) * 0.2
            End If

            'Die Werte der Zeilen 3 und 4 berechnen
            'lngTestPair(3) = lngTestPair(3) + (Cells(objCell.Row + 4, objCell.Column) - lngTestPair(3)) * 0.2
```

```
            'lngTestPair(4) = lngTestPair(4) + (Cells(objCell.Row + 3, objCell.Column) - lngTestPair(4)) * 0.4

        End If
        For i = 0 To 4
            Cells(objCell.Row - 4, objCell.Column) = lngTestPair(0)
            Cells(objCell.Row - 3, objCell.Column) = lngTestPair(1)
            Cells(objCell.Row - 1, objCell.Column) = lngTestPair(3)
            Cells(objCell.Row, objCell.Column) = lngTestPair(4)
        Next
        intChangeProcedure = intChangeProcedure + 1
        intTestCounter = 0
    End If
  Next


End Function

'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Bestimmt die Range Werte und trägt diese in einer zusätzlichen Spalte ein                                                '
'
'-----------------------------------------------------------------------------------------------------------------------------------------'
Sub RangeShow(rngData, strSheetName)
  Dim objCell As Range
  Dim SplitArray() As String
  Dim intRowCounter As Integer
  Dim intColumnCounter As Integer
  Dim intProcessCounter As Integer

  intRowCounter = 6
  intColumnCounter = 12
  intProcessCounter = 0
    'Es werde überprüfungen gemacht und die .. Werte gesplittet
    For Each objCell In rngData
        If InStr(objCell, "..") > 0 Then
            SplitArray() = Split(objCell, "..")
            While Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) <> ""
                If Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = "Range" > 0 Or
Sheets(strSheetName).Cells(intRowCounter, intColumnCounter + 1) = "" Then
                    GoTo InsertData
                Else
                    intColumnCounter = intColumnCounter + 1
                End If
            Wend
' Hier werden die Werte in die Hilfstabelle eingetragen
InsertData:
            Sheets(strSheetName).Cells(intRowCounter, intColumnCounter) = "Range"
            If intProcessCounter = 0 Or intProcessCounter = 4 Then
                Sheets(strSheetName).Cells(objCell.Row, intColumnCounter) = SplitArray(0)
            Else
                Sheets(strSheetName).Cells(objCell.Row, intColumnCounter) = SplitArray(1)
            End If
            objCell = SplitArray(0)
            intProcessCounter = intProcessCounter + 1

        End If
    Next
End Sub
```

```
'-----------------------------------------------------------------------------------------------------------------------------------------'
'Beschreibung: Dies ist die Hauptfunktion. Sie erstellt die Diagramme aus der Hilfstabelle.                                          '
'
'-----------------------------------------------------------------------------------------------------------------------------------------'
Sub TrendCreation(intObjectCounter, SheetName, intTestStepCounter, intTrendsToCreate, strHeader, strDiagSheetName,
intProcessCounter)
    Dim intObjects As Integer
    Dim ObjectName As Range
    Dim intDataRowCounter As Integer
    Dim ws As Worksheet
    Dim SheetExist As Boolean
    Dim ChartsCreated As ChartObject
    SheetExist = False
    intDataRowCounter = 0
    intRowCounter = 6
    intColumnCounter = 1
    intTestStepCounter = intTestStepCounter * 5
    SheetExist = False

    'Hier wird das Sheet für das Diagramm erstellt.
    For Each ws In Worksheets
        If ws.Name = strDiagSheetName & "_Diagram" Then
            Worksheets(ws.Name).Shapes.AddChart2(227, xlLine).Select
            Worksheets(ws.Name).ChartObjects("Diagramm " & intProcessCounter).Name = ws.Name & " " & intProcessCounter
            Worksheets(ws.Name).ChartObjects(ws.Name & " " & intProcessCounter).Activate
            SheetExist = True
            For Each ChartsCreated In Worksheets(ws.Name).ChartObjects
                If InStr(ChartsCreated.Name, ws.Name) > 0 Then

                Else
                    Sheets(ws.Name).ChartObjects(ChartsCreated.Name).Delete
                End If
            Next
        End If
    Next
    If SheetExist = False Then
        Set ws = ThisWorkbook.Worksheets.Add
        'Name wird zugeordnet
        ws.Name = strDiagSheetName & "_Diagram"
        Worksheets(ws.Name).Shapes.AddChart2(227, xlLine).Select
        Worksheets(ws.Name).ChartObjects("Diagramm " & intProcessCounter).Name = ws.Name & " " & intProcessCounter
        ActiveSheet.ChartObjects(ws.Name & " " & intProcessCounter).Activate
        SheetExist = False
    End If
    'Grösse und position wird angepasst
    With ActiveChart
        .Parent.Width = 250
        .Parent.Height = 200
        If intProcessCounter = 2 Or intProcessCounter = 4 Then
            .Parent.Left = 10
        Else
            .Parent.Left = 260
        End If
        If intProcessCounter = 3 Or intProcessCounter = 4 Then
            .Parent.Top = 10
        Else
            .Parent.Top = 220
        End If
        .HasTitle = True
        .ChartTitle.Text = strHeader
    End With
```

```
    'Die Zellen der Daten werden zugeordnet
    ActiveChart.SetSourceData Source:=Sheets(SheetName).Range(Sheets(SheetName).Cells(intRowCounter, intColumnCounter),
Sheets(SheetName).Cells(intRowCounter + intTestStepCounter, intColumnCounter + intObjectCounter)), PlotBy:=xlColumns

    'Die Achsen und Linien werden bearbeitet
    For i = 2 To intObjectCounter + 1
        If Sheets(SheetName).Cells(intRowCounter, intColumnCounter + i - 1) <> "" Or
InStr(Sheets(SheetName).Cells(intRowCounter, intColumnCounter + i - 1), "Datenreihen") = 0 Then
            ActiveChart.SeriesCollection(i).Name = Sheets(SheetName).Cells(intRowCounter, intColumnCounter + i - 1)
            'Bearbeitet die Achse
            If Sheets(SheetName).Cells(intRowCounter - 1, intColumnCounter + i - 1) = "Right" Then
                ActiveChart.SeriesCollection(i).AxisGroup = 2
            End If
            'Bearbeitet den LineType bzw. die Linie
            If Sheets(SheetName).Cells(intRowCounter - 4, intColumnCounter + i - 1) = "Pointed" Then
                ActiveChart.SeriesCollection(i).Select
                With Selection.Format.Line
                    .Visible = msoTrue
                    .DashStyle = msoLineSysDash
                End With
            End If
        End If
    Next

    'Hier werden alle unnötig zusätzlichen Datenreihen aus dem Diagramm gelöscht
    For i = 1 To 10
        On Error Resume Next
        intDataRowCounter = 0
        'Überprüfung auf "Datenreihen" und löschen
        If ActiveChart.SeriesCollection(i).Name = "" Or InStr(ActiveChart.SeriesCollection(i).Name, "Datenreihen") > 0 Or
ActiveChart.SeriesCollection(i).Name = "No" Then
            ActiveChart.SeriesCollection(i).Delete
            While intDataRowCounter < 10
                If InStr(ActiveChart.SeriesCollection(i).Name, "Datenreihen") > 0 Or ActiveChart.SeriesCollection(i).Name = "" Then
                    ActiveChart.SeriesCollection(i).Delete
                    intDataRowCounter = intDataRowCounter + 1
                Else
                intDataRowCounter = intDataRowCounter + 1
                End If
            Wend

        End If
    Next

    'Bearbeitung der XValues. Die Beschriftung mit den Testfällen
    ActiveChart.Axes(xlCategory).Select
    ActiveChart.SeriesCollection(1).XValues = Sheets(SheetName).Range(Sheets(SheetName).Cells(intRowCounter + 1,
intColumnCounter), Sheets(SheetName).Cells(intRowCounter + 1 + intTestStepCounter, intColumnCounter))


    'Legende hinzufügen
    ActiveChart.HasLegend = True

    intColumnCounter = 12
    intRowCounter = 6
    intObjects = intObjectCounter

    While Sheets(SheetName).Cells(intRowCounter, intColumnCounter) <> ""
        'Wenn eine False Spalte Existiert dann füge diese ein
```

```
        If Sheets(SheetName).Cells(intRowCounter, intColumnCounter) = "False" Or Sheets(SheetName).Cells(6, intColumnCounter)
= "Falsch" Then
            ActiveChart.SeriesCollection.NewSeries
            ActiveChart.SeriesCollection(intObjects + 1).Name = Worksheets(SheetName).Cells(intRowCounter, intColumnCounter)
            ActiveChart.SeriesCollection(intObjects + 1).Values =
Worksheets(SheetName).Range(Sheets(SheetName).Cells(intRowCounter + 1, intColumnCounter),
Sheets(SheetName).Cells(intRowCounter + intTestStepCounter, intColumnCounter))
            ActiveChart.SeriesCollection(intObjects + 1).Select
            'Gepunktet darstellen
            With Selection.Format.Line
                .Visible = msoTrue
                .DashStyle = msoLineSysDot
                .ForeColor.ObjectThemeColor = msoThemeColorBackground1
                .ForeColor.TintAndShade = 0
                .ForeColor.Brightness = -0.349999994
            End With
        'Wenn eine Range Spalte Existiert dann füge diese ein
        ElseIf Sheets(SheetName).Cells(intRowCounter, intColumnCounter = "Range") > 0 Then
            ActiveChart.SeriesCollection.NewSeries
            ActiveChart.SeriesCollection(intObjects + 1).Name = Worksheets(SheetName).Cells(intRowCounter, intColumnCounter)
            ActiveChart.SeriesCollection(intObjects + 1).Values =
Worksheets(SheetName).Range(Sheets(SheetName).Cells(intRowCounter + 1, intColumnCounter),
Sheets(SheetName).Cells(intRowCounter + intTestStepCounter, intColumnCounter))
            ActiveChart.SeriesCollection(intObjects + 1).Select
            'Gepunktet darstellen
            With Selection.Format.Line
                .Visible = msoTrue
                .DashStyle = msoLineSysDot
                .ForeColor.ObjectThemeColor = msoThemeColorBackground1
                .ForeColor.TintAndShade = 0
                .ForeColor.Brightness = -0.349999994
            End With
        End If
        intObjects = intObjects + 1
        intColumnCounter = intColumnCounter + 1
    Wend
    If intTrendsToCreate - 1 = 0 Then
        TrendCreator.Hide
        MsgBox "Trend Createt"
        Exit Sub
    End If
End Sub
```