



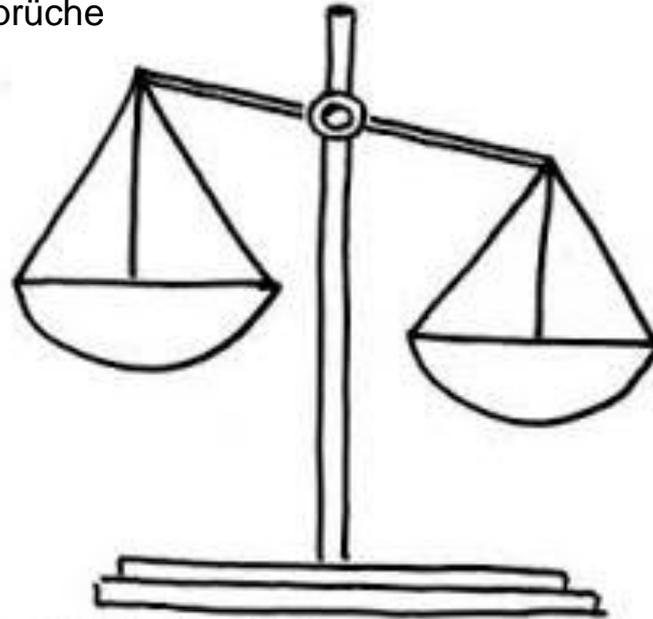
Mut zur Lücke

Gebhard Ebeling, Mario Krahrmer

arvato
BERTELSMANN

FINANCIAL SOLUTIONS

- Höchste Qualitätsansprüche
- Bester am Markt
- Erstklassige Software



- Termindruck
- fehlende Spezifikationen
- knappe Ressourcen
- begrenzte Budgets

- Qualität ja... aber zu welchem Preis?

- Was passiert wenn die Qualität nicht ausgereicht hat?

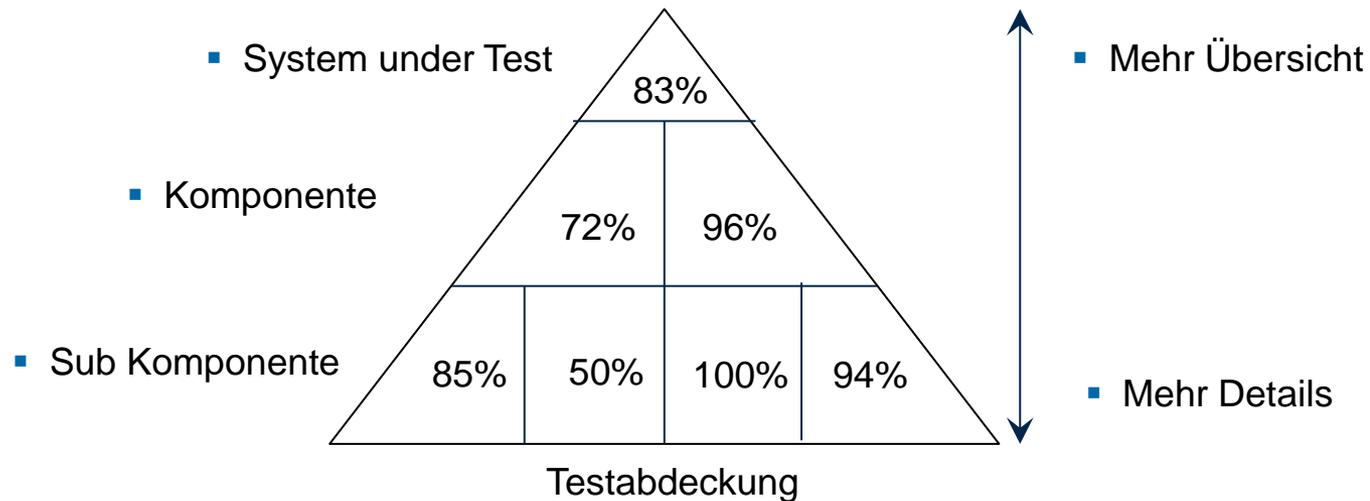
Warum wurde das nicht getestet?

- Wann wird die Frage gestellt? *Wenn Kunden sich beschweren (zu spät)*
- Was war die häufigste Antwort? *Keine Zeit, kein Budget, nicht spezifiziert, ...*
- Wer kann das beeinflussen? *Projektleiter, Product Owner, Testmanager*
- Ziel: Ein gemeinsames Abwägen zwischen Qualitätsmaßnahmen, Termin und Budget ermöglichen

Wie kommen wir weiter?

- Aufgrund des Termindrucks und fehlender Mittel ist ein vollständiger Test oft nicht möglich
- Das dadurch verbleibende Restrisiko muss für alle transparent dargestellt werden um eine Entscheidungsgrundlage / Diskussionsgrundlage zu haben
- Wichtig: Was wird getestet?
- Wichtiger: Was wird nicht getestet?
- Wir brauchen einen Weg vom Fingerprinting zu einer gemeinsamen Risikoabschätzung
 - Wie können wir „Qualität“ messbar darstellen?

Wie können wir „Qualität“ messbar darstellen?

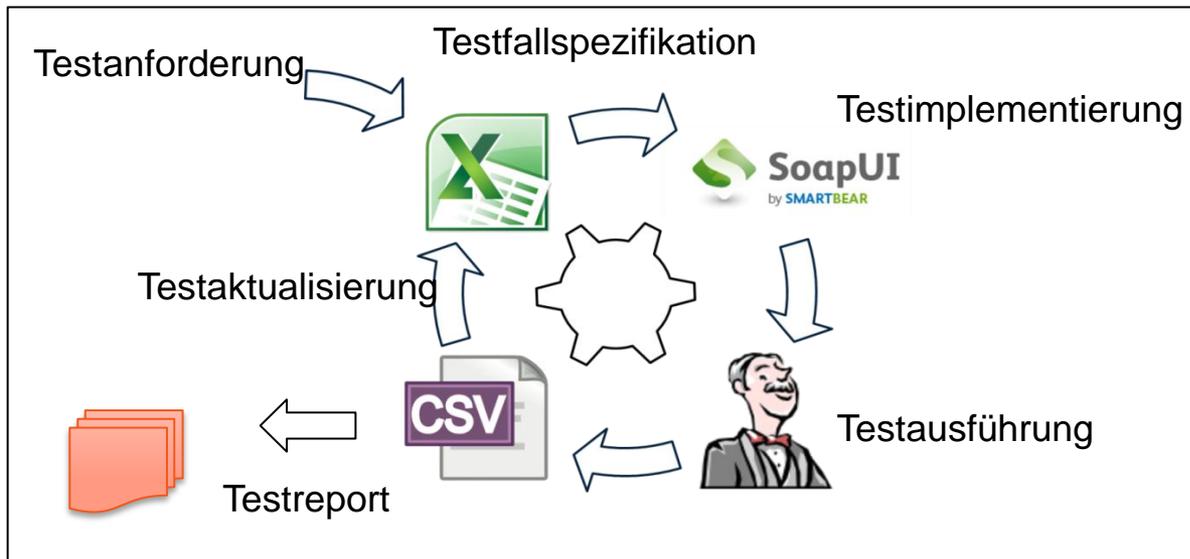


- Wie kann eine praktische Umsetzung aussehen?

Ausgangslage:

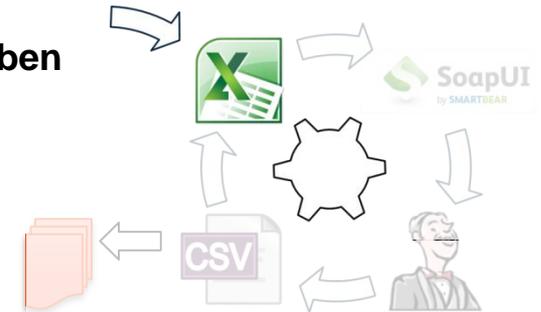
- Unvollständige Systemdokumentation
- Unvollständige Testdokumentation
- Unterschiedliche Testtools im Einsatz
- Testreport(s) nicht belastbar / aussagekräftig genug für Produktionsfreigabe

Problemlösung durch Umsetzung des Testkreislaufs



1. Testfälle werden in einer separaten Testspezifikation beschrieben

- Standard Merkmale von Tests
- Jeder Testfall hat eine eindeutige ID
- Testfälle sind zielgerichtet und prüfen nur das erwartete Ergebnis



ID	Name	Description	Precondition	Expected Result	Comment	Status
TOP_TEP-0001	Testname	- Kurze aber Aussagekräftige Beschreibung des Tests, die sein Ziel enthält	- Alle benötigten Voraussetzungen für die erfolgreiche Testausführung	- Erwartendes Testverhalten	- Zusätzliche Informationen - Z.B. Wieso kann der Test nicht implementiert werden - Z.B. Wieso wurde der Test gelöscht	- Planned - Designed - Implemented - Known Failed - Disabled - Deleted

2. Testfälle haben einen Status

- **Implemented**
Es gibt eine Implementierung zu einem Test
- **Designed**
Ein Test ist vorgesehen und beschrieben
- **Planned**
Ein Test ist vorgesehen aber nicht beschrieben

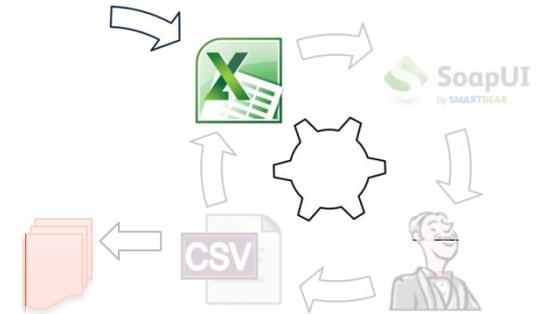


ID	Name	Description	Precondition	Expected Result	Comment	Status
TOP_TEP-0001	Testname	- Kurze aber Aussagekräftige Beschreibung des Tests, die sein Ziel enthält	- Alle benötigten Voraussetzungen für die erfolgreiche Testausführung	- Erwartendes Testverhalten	- Zusätzliche Informationen - Z.B. Wieso kann der Test nicht implementiert werden - Z.B. Wieso wurde der Test gelöscht	- Planned - Designed - Implemented - Known Failed - Disabled - Deleted

Warum Status Planned?

(Ein Test ist vorgesehen aber nicht beschrieben)?

- Zeitaspekt
- Wissen fehlt



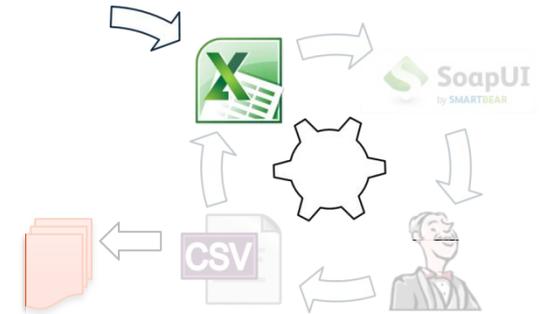
Effekt:

- Testabdeckung wird geringer
 - Genauigkeit der Qualitätsaussage steigt
 - **10 geplante Tests sind besser als eine nicht dargestellte Testlücke**

ID	Name	Description	Precondition	Expected Result	Comment	Status
TOP_TEP-0001	Testname	- Kurze aber Aussagekräftige Beschreibung des Tests, die sein Ziel enthält	- Alle benötigten Voraussetzungen für die erfolgreiche Testausführung	- Erwartendes Testverhalten	- Zusätzliche Informationen - Z.B. Wieso kann der Test nicht implementiert werden - Z.B. Wieso wurde der Test gelöscht	- Planned - Designed - Implemented - Known Failed - Disabled - Deleted

3. Testspezifikation unabhängig von der Testimplementierung

- Toolfreiheit
- Scrum Team und PO beschreiben Testfälle ohne Rücksicht auf Umsetzungsmöglichkeiten
- Testable Requirements als Antwort auf fehlende Produktspezifikation ist die Basis der Testabdeckung
- Testdesign mittels Testmethoden (Aquivalenzklassen, Pairwise, etc)

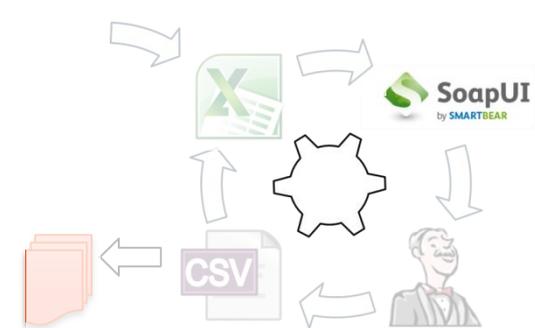


ID	Name	Description	Precondition	Expected Result	Comment	Status
TOP_TEP-0001	Testname	- Kurze aber Aussagekräftige Beschreibung des Tests, die sein Ziel enthält	- Alle benötigten Voraussetzungen für die erfolgreiche Testausführung	- Erwartendes Testverhalten	- Zusätzliche Informationen - Z.B. Wieso kann der Test nicht implementiert werden - Z.B. Wieso wurde der Test gelöscht	- Planned - Designed - Implemented - Known Failed - Disabled - Deleted

Testtool kann frei gewählt werden, mit folgender Einschränkung:

Toolunabhängige Reporting Schnittstelle (z.B. CSV Format)

- SoapUI
 - Umsetzung durch Groovy Erweiterung
- JUnit
 - Umsetzung durch Annotation
- Skripte wie Python, Perl, Batch
 - Umsetzung trivial

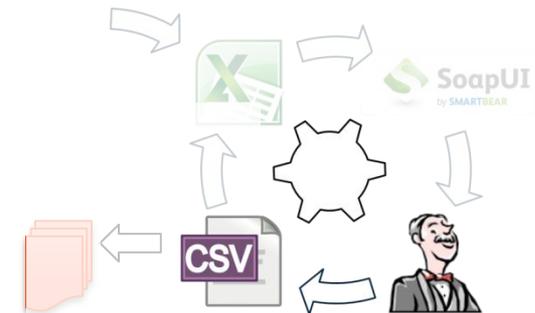


Bezug zur Testspezifikation über die Testfall ID

Result	Start Time	Duration	Location	Failed Reason
- Status der letzten Ausführung	- Startzeit der letzten Ausführung	- Laufzeit des Tests	- Wo ist der Test implementiert	- Detailliertes Fehlermeldung beim Fehlschlagen

Continuous Integration System (z.B. Hudson)

- Ausgabe: Report Dateien mit
 - Testergebnissen für den Testreport
 - Spezifikationsänderungen zur Testaktualisierung

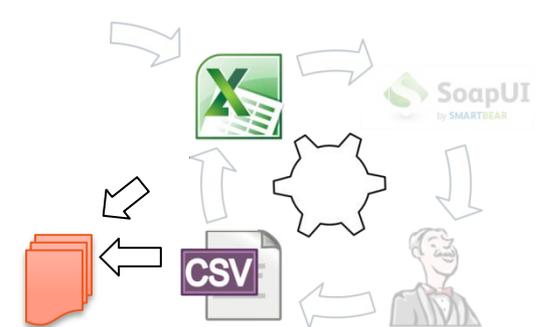


Result	Start Time	Duration	Location	Failed Reason
- Status der letzten Ausführung	- Startzeit der letzten Ausführung	- Laufzeit des Tests	- Wo ist der Test implementiert	- Detaillierte Fehlermeldung beim Fehlschlagen
- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert.

Release Test

- Testspezifikation (Soll) + Report Dateien (Ist) = Testreport

Teil der Releasedokumentation

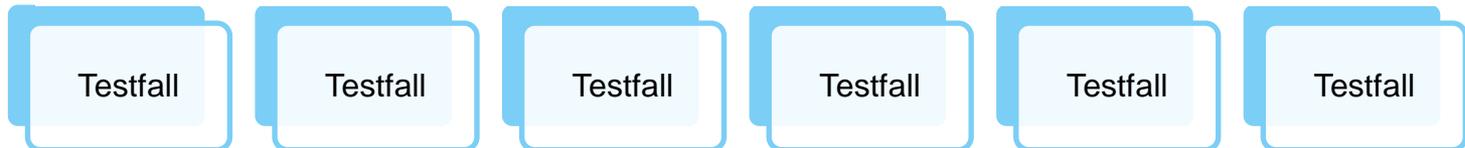


- Testreporterstellung erfolgt automatisiert

Result	Start Time	Duration	Location	Failed Reason
- Status der letzten Ausführung	- Startzeit der letzten Ausführung	- Laufzeit des Tests	- Wo ist der Test implementiert	- Detaillierte Fehlermeldung beim Fehlschlagen
- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert	- Wird vom Testwerkzeug generiert.

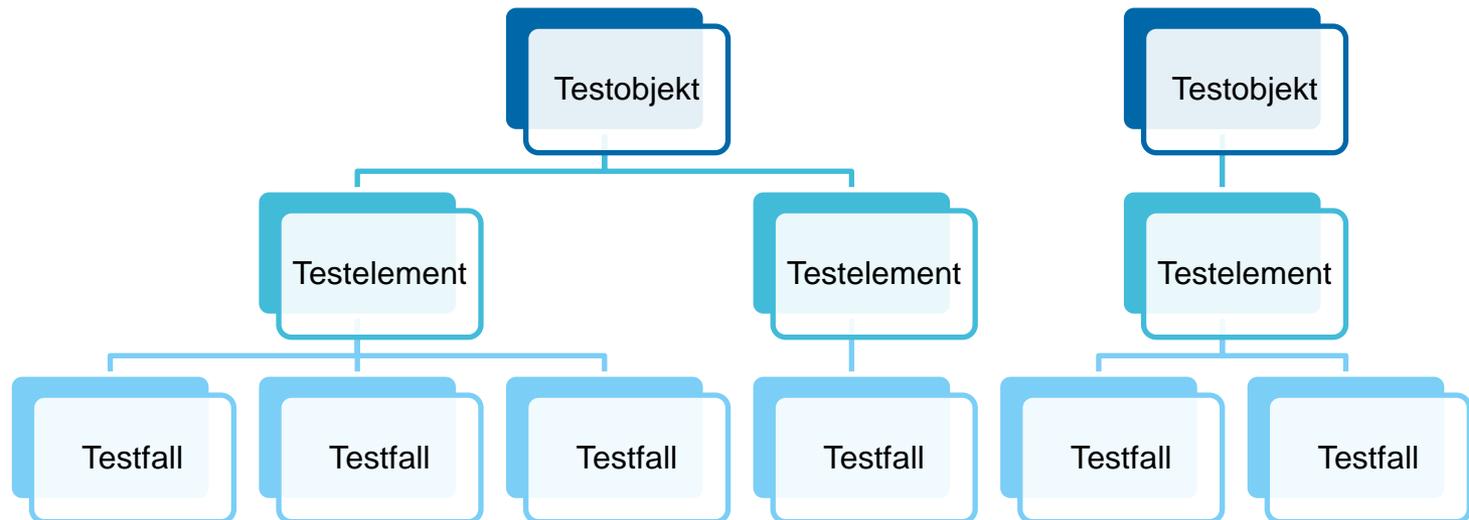
Viele Testreports -> Qualitätsaussage?

- Alle Testfälle (durchgeführt, implementiert, spezifiziert, geplant)
- Keine Übersicht
- Qualifizierte Aussage noch nicht möglich



Strukturierung

- Testobjekt - Die Komponente, welche getestet wird
- Testelement – Funktionale Einheit
- Testfall



Alle Tests

Test mit Status: implementiert, spezifiziert, geplant

Spezifizierte Tests

Test ist vollständig beschrieben
(implementiert, spezifiziert, ~~geplant~~)

Implementierte Tests

Es gibt eine Implementierung zu
dem spezifizierten Test
(implementiert, spezifiziert, ~~geplant~~)

Implementierungsgrad

Anzahl Implementierten Tests / Anzahl Alle Tests

Durchgeführte Tests

Anzahl der implementierten
durchgeführten Tests

Fehlgeschlagene Tests

Anzahl der Tests mit einem Fehler

Bekannte Fehlgeschlagene Tests

Tests die ein dokumentiertes Fehlverhalten
aufzeigen

Funktionelle Testabdeckung

Anzahl Durchgeführter Tests / Anzahl Aller
vorgesehenen Tests

Metriken eines Testobjekt

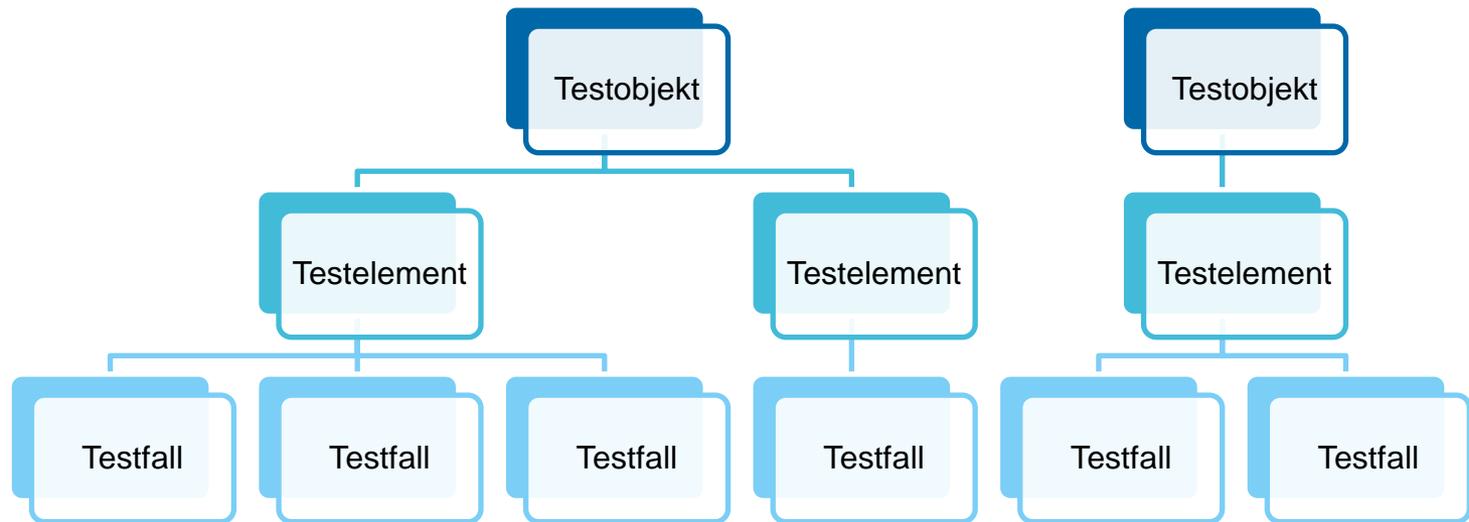
Testobjekt:	Neues Produkt
Ansprechpartner	Scrum Team 1
Testobjekt Präfix	B_FC_C
Beschreibung:	Lösch Batch Job für abgelaufene Produkt Daten
Getestete Version	13.02.001
Umgebung	DEV
Alle Tests	1234
Summe Spezifizierte Tests:	1048
Summe Implementierte Tests:	978
Implementierungsgrad	79%
Anzahl Review Findings	13
Durchgeführte Tests	865
Fehlgeschlagene Tests	13
Bekannte Fehlgeschlagene Tests	13
Testabdeckung	70%

Testfallspezifikation

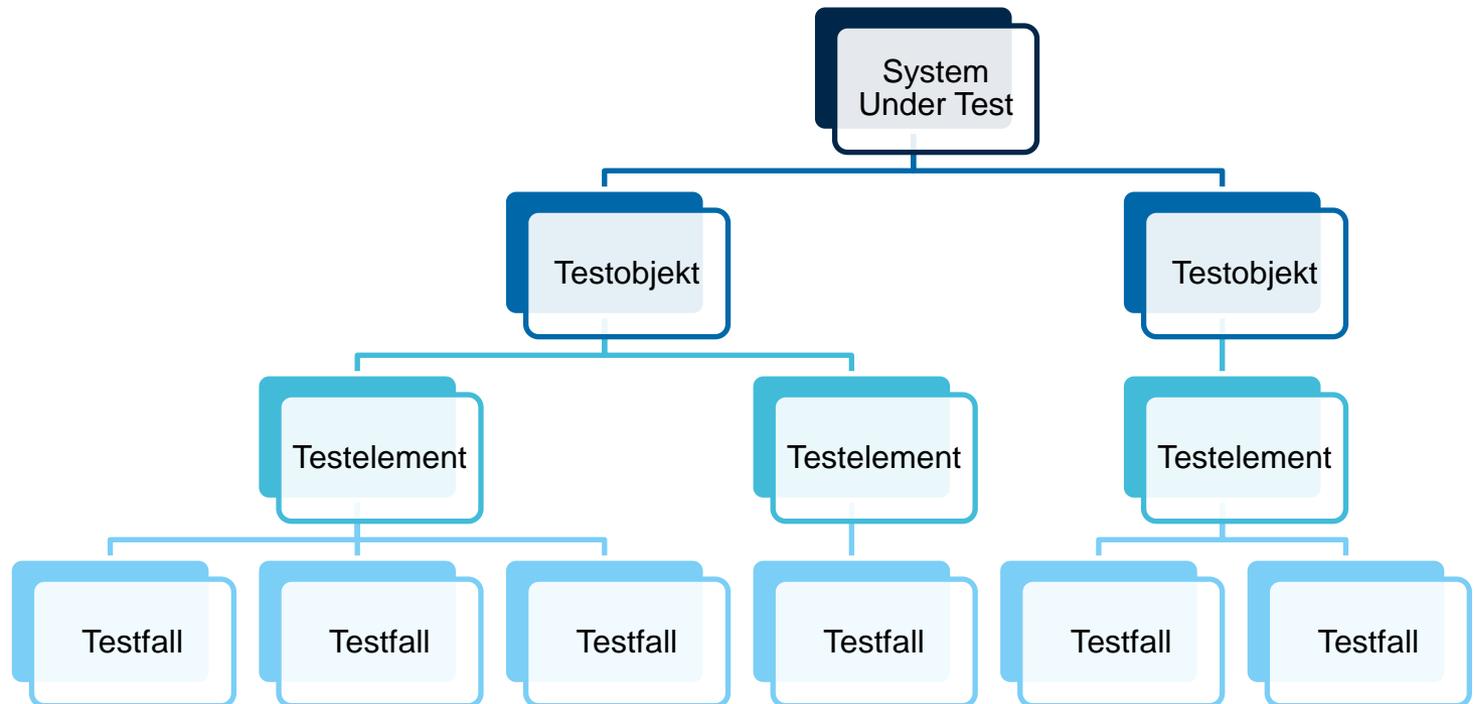
Testobjekt:	Neues Produkt
Ansprechpartner	Scrum Team 1
Testobjekt Präfix	B_FC_C
Beschreibung:	Lösch Batch Job für abgelaufene Produkt Daten
Getestete Version	13.02.001
Umgebung	DEV
Alle Tests	1234
Summe Spezifizierte Tests:	1048
Summe Implementierte Tests:	978
Implementierungsgrad	79%
Anzahl Review findings	13
Durchgeführte Tests	865
Fehlgeschlagene Tests	13
Bekannte Fehlgeschlagene Tests	13
Testabdeckung	70%

Testelement Präfix	Arbeitsblatt	Name	Beschreibung	Alle Tests	Spezifizierte Tests	Implementierte Tests	Implementierungsgrad	Anzahl Review Findings	Durchgeführte Tests	Fehlgeschlagene Tests	Bekannte Fehlgeschlagene Tests	Testabdeckung
FUNC	B_FC_C_FUNC - Funktionell	Batchfunktion	Funktionale Anforderungen werden getestet	1189	1003	933	78%	13	820	13	13	68%
NFUNC	B_FC_C_NFUNC - Nicht Funktionell	Performance	Nichtfunktionale Anforderungen werden getestet	45	45	45	100%	0	45	0	0	100%
Gesamt				1234	1048	978	79%	13	865	13	13	70%

System besteht aus mehreren Testobjekten



System besteht aus mehreren Testobjekten



- Qualitätssicherung ist messbar
- Planned / Designed Tests zur Ermittlung der Testbasis nutzen
- Der Testumfang wird allen Verantwortlichen verständlich und transparent dargestellt
- Es können gezielt Maßnahmen zur Qualitätsverbesserung eingeleitet werden
- Eine Releasefreigabe kann entschieden werden

Die Qualitätsverantwortung kann von allen getragen werden.

Thanks!

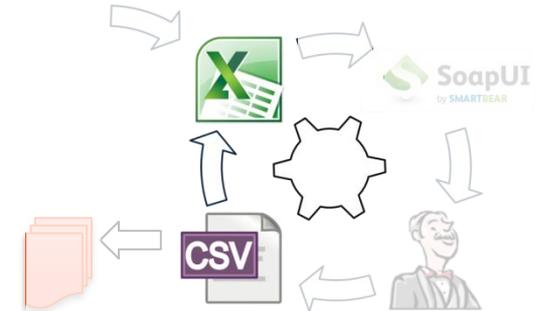
Gebhard Ebeling
Testmanager
gebhard.ebeling@bertelsmann.de

Mario Kraemer
Testmanager
mario.kraemer@bertelsmann.de

arvato Financial Solutions
Rheinstraße 99
76532 Baden-Baden
finance.arvato.com

Entwicklung

- Zur Akzeptanz
 - Vermeidung Bürokratie
 - Reverse Engineering
-
- Veränderung der Spezifikationsmerkmale aus den Testtools automatisch in die Spezifikation zurückführen
(z.B. Beschreibung, Erwartetes Ergebnis)



ID	Name	Description	Precondition	Expected Result	Comment	Status
TOP_TEP-0001	Testname	- Kurze aber Aussagekräftige Beschreibung des Tests, die sein Ziel enthält	- Alle benötigten Voraussetzungen für die erfolgreiche Testausführung	- Erwartendes Testverhalten	- Zusätzliche Informationen - Z.B. Wieso kann der Test nicht implementiert werden - Z.B. Wieso wurde der Test gelöscht	- Planned - Designed - Implemented - Known Failed - Disabled - Deleted

Verbesserung mit Reviews

Anzahl Review Findings

Regelmäßige Reviews verbessern die Testbeschreibungqualität

- Was ist das Ziel des Testes
- Ist die Formulierung verständlich
- Hält sich der Test an vorhandene Vorgaben

Anmerkungen können an jeden Tests geschrieben werden

Zählen aller Anmerkungen

Je mehr Anmerkungen -> Eine Verbesserung sollte eingeplant werden

Quelle ISTQB Glossar

- Qualität = Der Grad, in dem ein System, eine Komponente oder ein Prozess die Kundenerwartungen und -bedürfnisse erfüllt. [Nach IEEE 610] (2) Der Grad, in dem ein Satz inhärenter Merkmale Anforderungen erfüllt. [ISO 9000:2000]
- Risiko = Ein Faktor, der zu negativen Konsequenzen in der Zukunft führen könnte; gewöhnlich ausgedrückt durch das Schadensausmaß und die Eintrittswahrscheinlichkeit.
- Risikoanalyse = Bewertung von identifizierten Projektrisiken oder Produktrisiken um ihre Risikostufe zu bestimmen, typischerweise durch die Bewertung von Schadensausmaß und Eintrittswahrscheinlichkeit.