

SOFTWAREVALIDIERUNG / COMPUTER SOFTWARE ASSURANCE

Vereinfachte Dokumentation für flexible Softwarevalidierung

22.09.2025

www.testotis.de

AGENDA



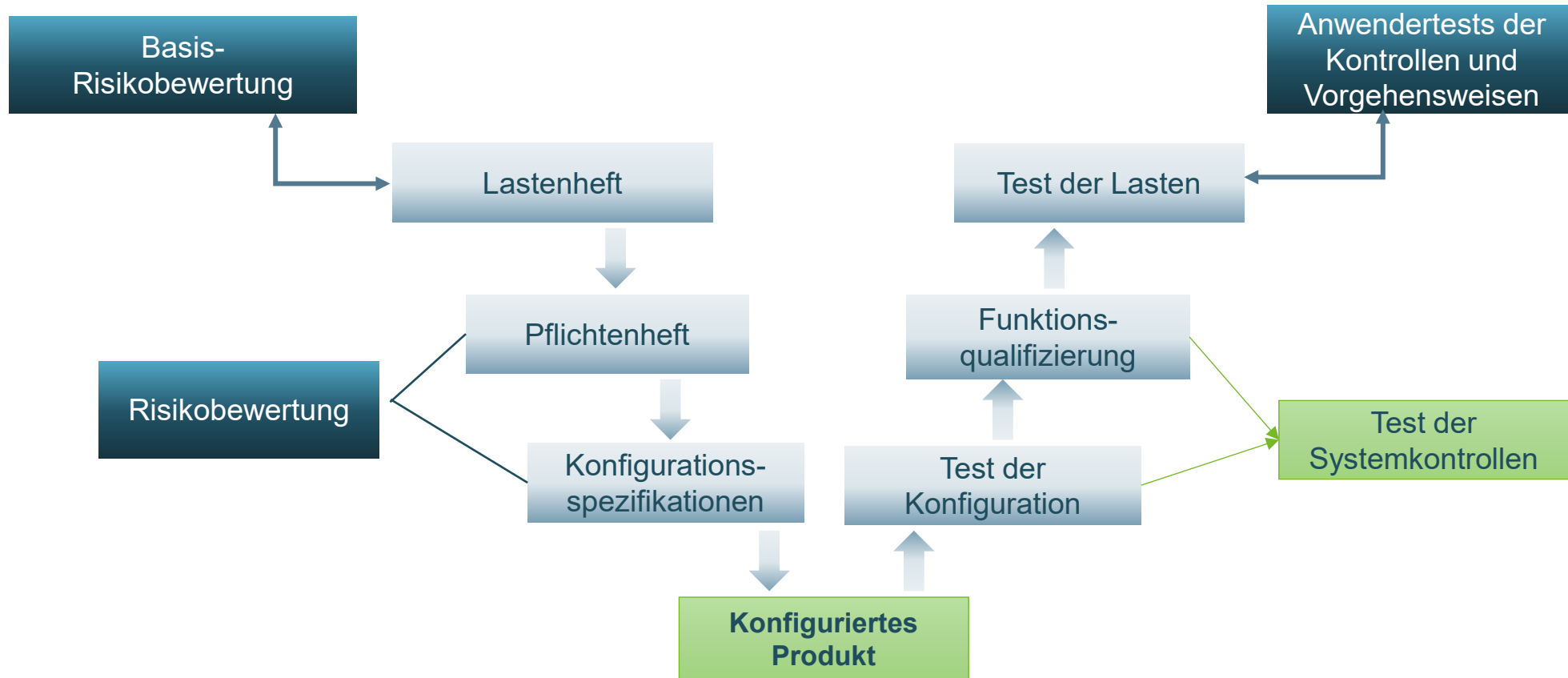
- ▶ **COMPUTER SOFTWARE ASSURANCE**
- ▶ **RISIKOBEWERTUNG**
- ▶ **TESTUNG UND TESTAKTIVITÄTEN**
- ▶ **OPTIMIERTER VALIDIERUNGSANSATZ**

Regularien und Leitlinien



- ▶ 21 Code of Federal Regulations (CFR) **Part 820** – Quality System Regulation
- ▶ 21 Code of Federal Regulations (CFR) **Part 11** – Electronic Records; Electronic Signatures
- ▶ DIN EN ISO **13485:2016** – Medizinprodukte – Qualitätsmanagementsysteme – Anforderungen für regulatorische Zwecke
- ▶ EU GMP-Leitfaden **Anhang 11** – Computerised Systems
- ▶ **ISPE GAMP®5** – A Risk-Based Approach to Compliant GxP Computerized Systems und weitere ISPE GAMP Good Practice Guides
- ▶ **ISO/TR 80002-2** – Validation of software for medical device quality systems

Klassische CSV im V-Modell (GAMP-Kategorie 4)



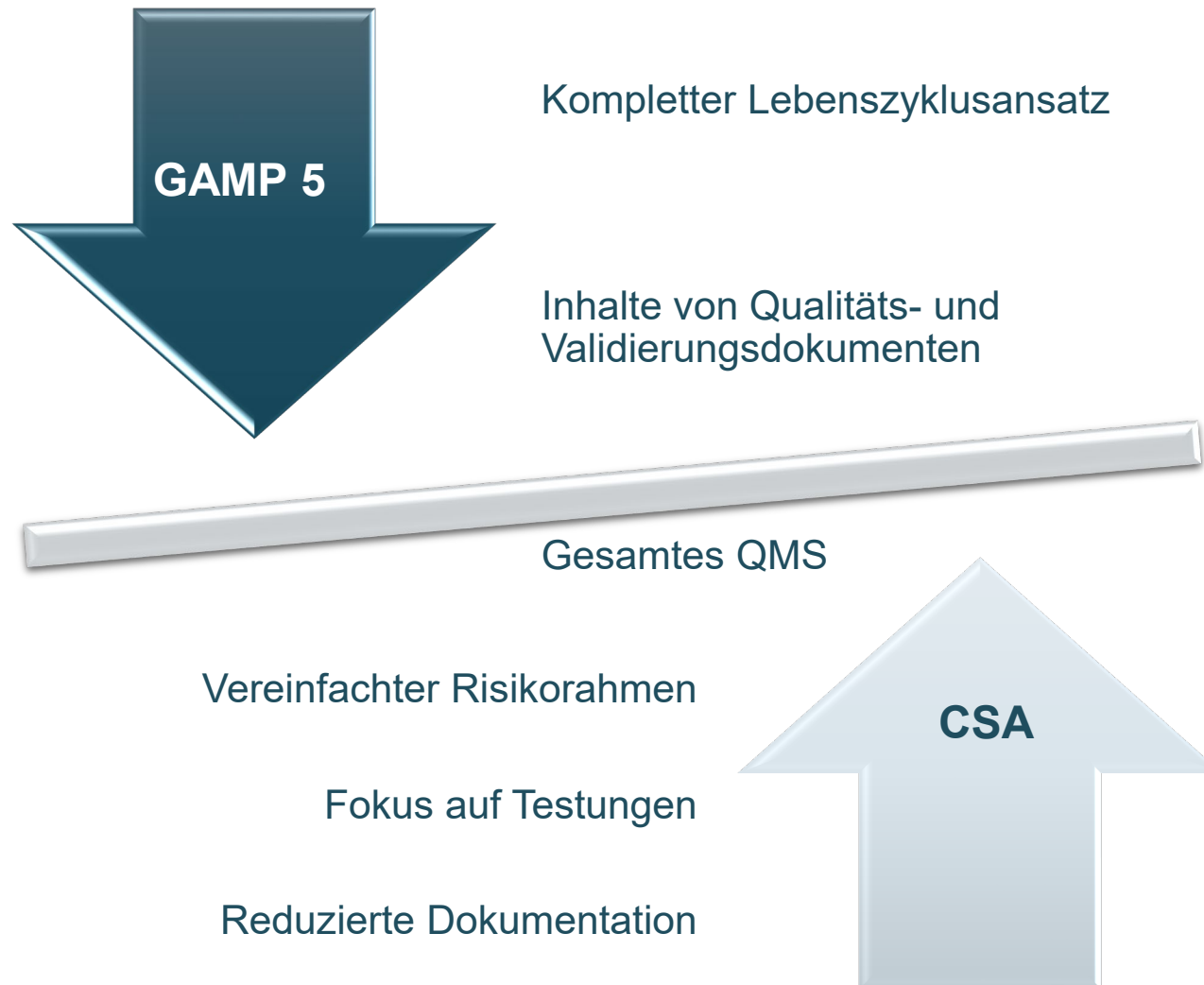
COMPUTER SOFTWARE ASSURANCE



- ▶ Optimierung von Sicherheit und Qualität bei der Softwarevalidierung für regulierte Anwender
- ▶ Verschlinkung und Rationalisierung des Validierungsansatzes durch risikobasiertes und kritisches Denken
- ▶ Unterscheidung von Softwarefunktion mit hohem und niedrigem Prozessrisiko gemäß beabsichtigtem Einsatzzweck

WARUM COMPUTER SOFTWARE ASSURANCE?

Softwarekategorien



Risikorahmen



1. Verwendungszweck

- Teil eines Qualitätssicherungs- oder Produktionssystems
- Validierungspflicht

2. Risiko

- Geeignete Risikoanalyse
- Risiken für Patientensicherheit, Produktqualität, Datenintegrität

3. Sicherstellungsaktivitäten

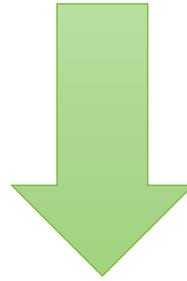
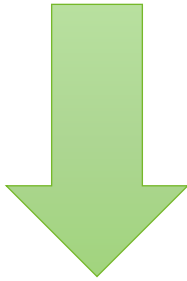
- Angemessene Maßnahmen, um Fehlerauswirkung auf akzeptables Niveau zu reduzieren
- z. B. Testen, zusätzliche Prüfungen im weiteren Verlauf des Prozesses, ergänzende Arbeitsanweisungen und/oder Schulungen

4. Art und Weise der Aufzeichnungen

Wann muss validiert werden?



Validierungspflichtig



DIREKT:
Teil des Produktion- oder
QM-Systems

INDIREKT:
Unterstützung des
Produktion- oder QM-
Systems

Nicht Validierungspflichtig



Infrastruktur, die nicht
produktions-/
qualitätsspezifisch ist
oder
Verwaltung allgemeiner
Geschäftsprozesse

RISIKOBASIERTES VORGEHEN



Risikobewertung CSA



- ▶ Risikobasierte Analyse für SW-Features, -Funktionen, -Bedienungen, die Teil der Produktion / des QMS sind
- ▶ Identifikation von hohen Prozessrisiken bei Softwareausfall/-fehler
- ▶ Faktoren, die den beabsichtigten Zweck beeinflussen:
 - ordnungsgemäße Systemkonfiguration und -verwaltung,
 - Sicherheit des Systems,
 - Datenspeicherung,
 - Datenübertragung oder Betriebsfehler

Ungleich bei Software als Medizinprodukt gemäß DIN 14971 : Auftrittswahrscheinlichkeit kann nicht auf historischen Daten oder Modellierung geschätzt werden

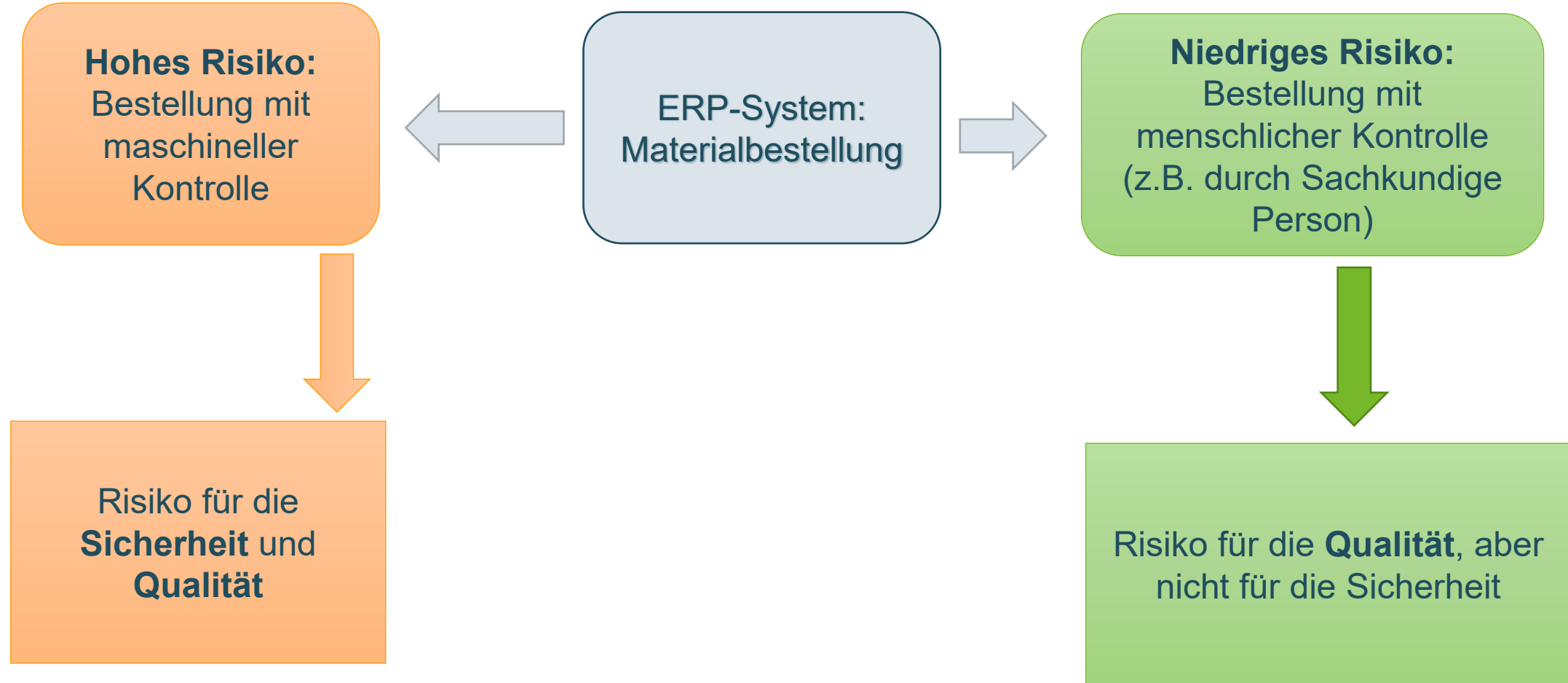
Prozessrisiken

Potential, die Produktion / das Qualitätssystem zu beeinträchtigen

Medizinprodukte-Risiken

Potential, eines Produkts die Patienten-/ Anwendersicherheit zu beeinträchtigen
→ resultierend aus einem Qualitätsproblem

Beispiel für Einteilung von Risiken



Risikoklassifizierung



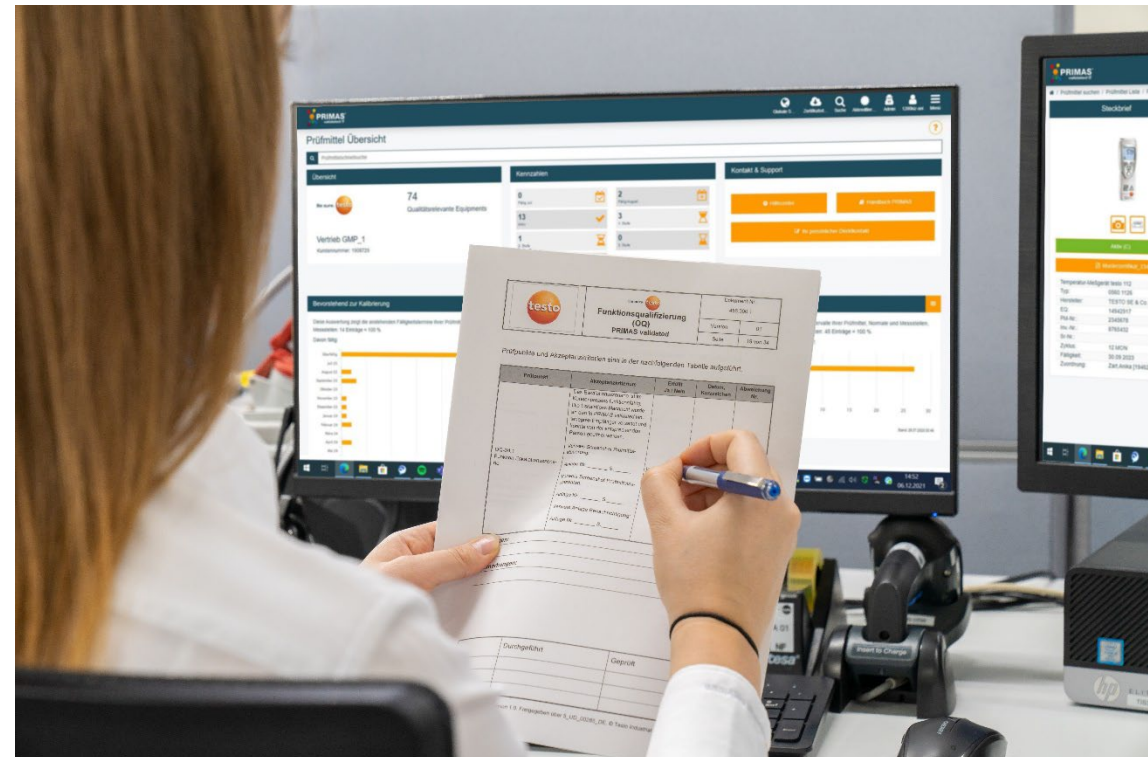
Zwei Risikofaktoren:

Patienten- einfluss	direkt	niedrig	mittel	hoch
	indirekt	unkritisch	niedrig	mittel
GAMP-Kategorie (Komplexität)		3	4	5

Risiko-
kategorie

Risikokategorie ergibt die
risikoreduzierenden Maßnahmen und
den Testaufwand

TESTUNGEN / TESTAKTIVITÄTEN



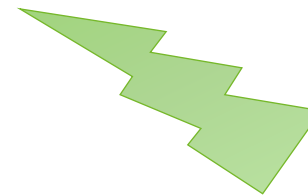
Reduzierung unnötiger Testaktivität

- ▶ Ungeskripte bzw. limitiert geskripte Tests
- ▶ Bereits bestehende Kontrollen und Mechanismen nutzen
- ▶ Risikobasiertes Testen

Tests, welche der Anbieter oder Entwickler der Software bereits durchgeführt und hinreichend dokumentiert hat.



Herstellerdokumentation



Lieferantenqualifizierung / Audits

Ungeskriptet

- Kein Testskript
- Keine/Rudimentäre Testanweisung
- Keine erwarteten Ergebnisse
- Testung wird dokumentiert
- Keine Testnachweise, außer bei Abweichungen
- Keine Vorab-Freigabe

Limitiert geskriptet

- Reduzierte Testanweisung
- Dokumentierter Testnachweis nur für finales Ergebnis
- Keine Vorab-Freigabe

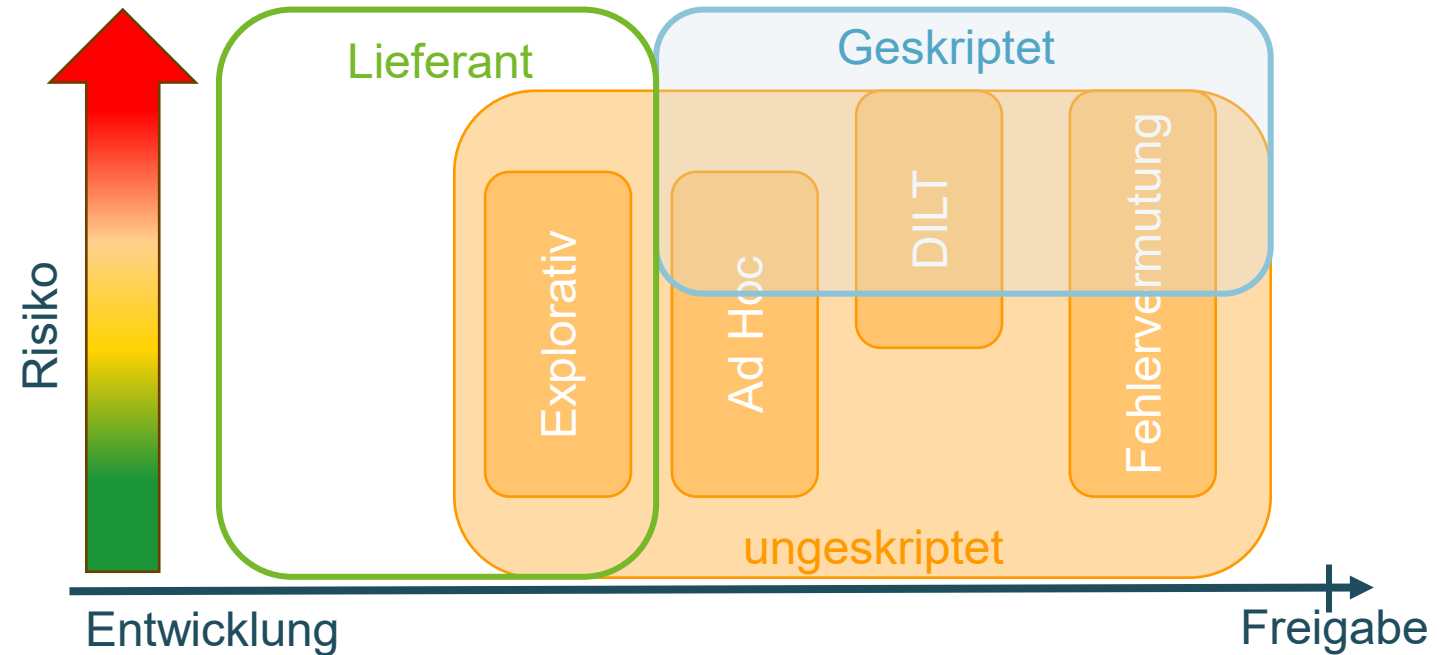
Robust geskriptet

- Schritt-für-Schritt-Testanweisung
- Erwartetes Ergebnis
- Dokumentierter Testnachweis pro Schritt
- Vorab-Freigabe des Skripts

Ungeskriptetes Testen

Dynamisches Testen, bei dem die Aktionen des Testers nicht durch schriftliche Anweisungen in einem Testfall vorgeschrieben sind. Es umfasst:

- ▶ Ad-hoc-Tests
 - Tests ohne Dokumentation
- ▶ Fehlervermutung
 - Tests auf Grundlage früherer Fehler
- ▶ Exploratives Testen
 - Erfahrungsbasiertes Testen
 - zufällige Benutzungssituationen
- ▶ Day in the life testing (DILT)



Darstellung gemäß GAMP® 5



GAMP® 5 liefert Möglichkeit für den Einsatz und die Dokumentation von ungeskripteten Tests (auf für agile Ansätze oder SaaS)

Vergleich: Ungeskriptetes vs. Geskriptetes Testen



Ungeskriptet	Geskriptet
Fehleridentifikation wird erhöht	Testung von spezifischen Funktionen mit hohem GxP-Risiko
Aufdeckung von Fehlern auch z. B. durch ungenügend definierte Spezifikationen	Testung anhand von Spezifikationen und Bestätigung der Eignung für beabsichtigten Einsatzzweck
Basiert auf Wissen/Erfahrung des Testers	Die Spezifikation bildet den Standard
Testung von erwartetem und unerwartetem System- / Nutzerverhalten	Testung des erwarteten System- / Nutzerverhalten

Klassische Testdokumentation



Test Fall Nr.	Funktion gemäß Lastenheft	Test-schritt Nr.	Detaillierte Beschreibung der Testaktivität	Akzeptanzkriterien	Ergebnis	Akzeptanz-kriterium erfüllt? [J/N]
2	7.3.1 Datenerfassung Artikelnummer	1	Aufruf des Menüpunkts „Dateneingabe Artikelnummer“	Eingabefeld „Artikelnummer“ erscheint		
		2	Eingabe von „A047110E“	Eingabe wird mit Fehlermeldung „Prüfsumme Artikelnummer nicht korrekt“ abgelehnt		
		3	Eingabe von „A041170E“	Eingabe wird akzeptiert und Daten zu Artikel „Testmaterial“ erscheinen		

Datum/Kürzel Tester_____

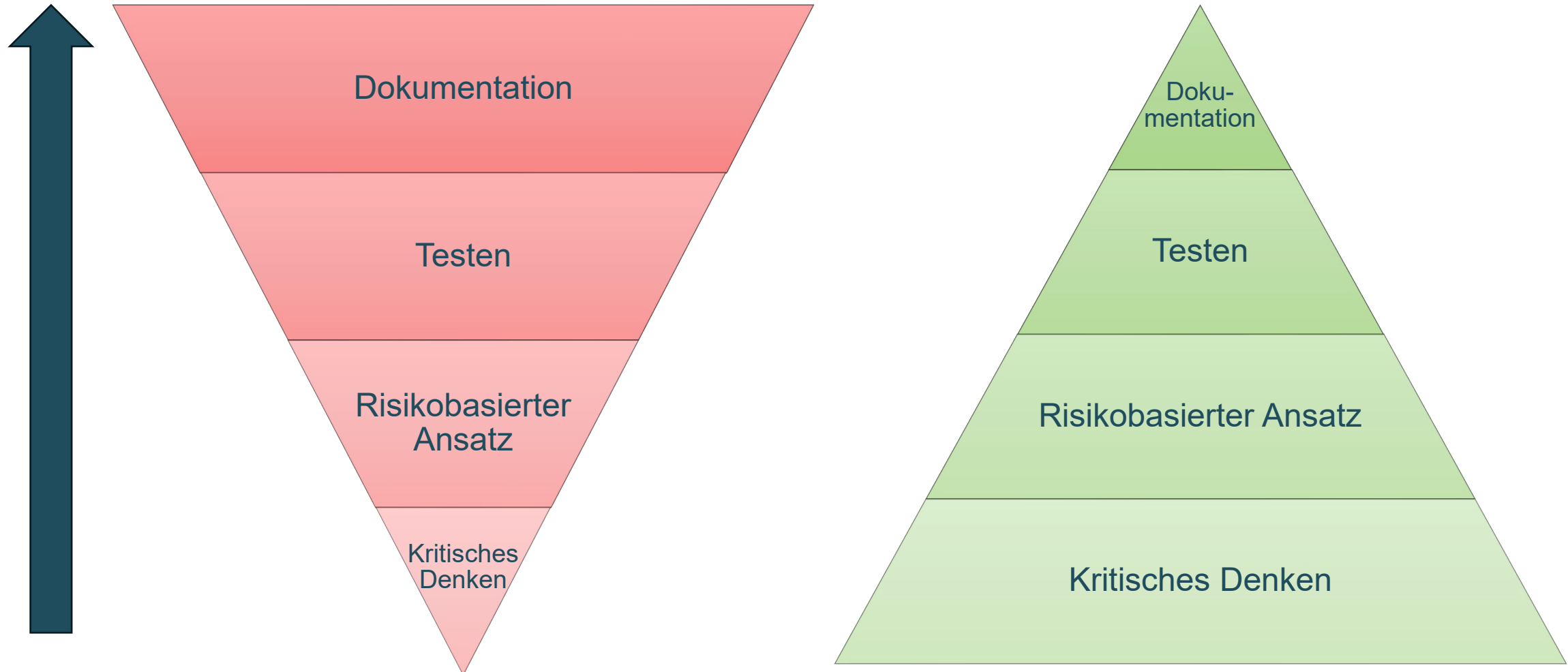
Anlagen (Screenshots, Ausdrücke):_____

Ungekripte Testdokumentation



Ziel des Tests	Testergebnis	Bemerkung und/oder Verweis auf andere Testdokumentation
	☐ Bestanden (keine Abweichung oder Nachweis durch Herstellertestung) ☐ Nicht bestanden, Abweichung-ID:	
Genehmigung der Testergebnisse (der oben genannte Testumfang wurde vollständig getestet und alle Abweichungen vollständig dokumentiert)		
Name (Tester)	Datum	Kürzel

Optimierung des Validierungsansatzes



Beispiel E-Signaturen im Abweichungsmanagement



Funktionen:

- Zwei Komponenten
- Aufzeichnung im Audit Trail
- Informationen über Aufzeichnung:
 - Name
 - Bedeutung
 - Datum, Uhrzeit

Verwendungszweck der Software:

Normgerechte Erfassung und Aufbewahrung elektronischer Unterschriften

Risikoanalyse:

- Bei Ausfall des Systems:
 - Kein Nachweis von Genehmigungen
 - Mangelhafte Auditierbarkeit
 - Evtl. Verletzung von Normen
- Qualitäts- aber kein Sicherheitsrisiko

Sicherungsmaßnahmen:

- Bewertung der Systemkapazität, Lieferantenqualifizierung IQ,
 - Ad-Hoc Testungen mit Anwendern
- Nachweis der richtigen Funktionsweise für den Einsatzzweck

Niedriges Risiko

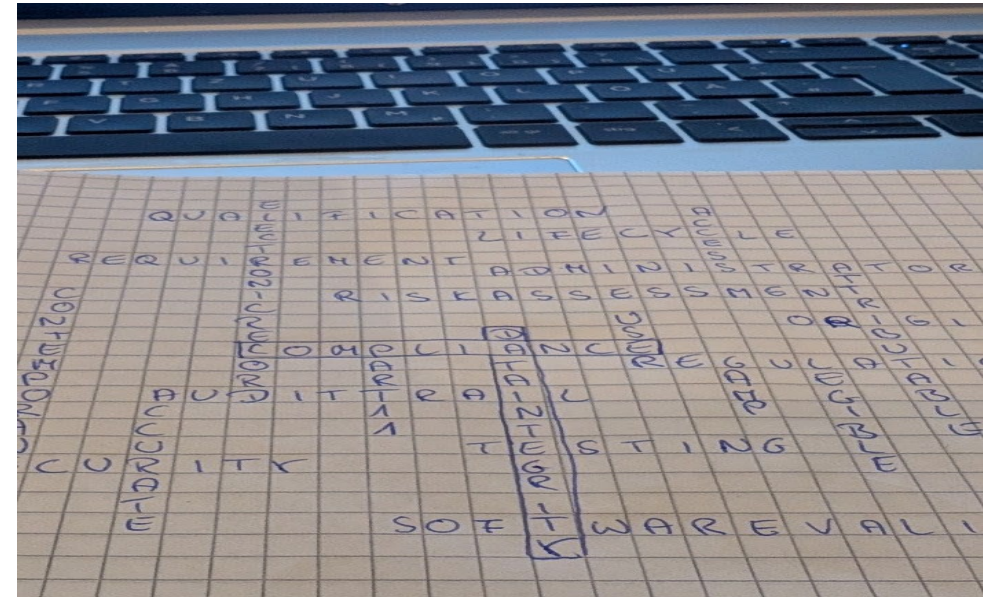
Dokumentation:

- Verwendungszweck
- Risikoanalyse
- Ausgeführte Tests und deren Ergebnis
- Aussage über die Akzeptanz der Software
- Datum und Ausführer der Tests

Fazit zum verbesserten Ansatz



- ▶ Anwendung von kritischem Denken und Fokus der Validierung auf die hohen Risiken
- ▶ Ganzheitliche Risikobetrachtung mit Fokus auf Patientensicherheit, Produktqualität und kritische Datenintegrität.
- ▶ Keine detaillierten Testskripte für Prozesse/ Funktionen mit niedrigerem Risiko
- ▶ Weniger Zeit notwendig für die
 - Erstellung von Dokumenten
 - Testdokumentation/Nachweise
 - Korrektur von Fehlern in Testskripten
- ▶ Dafür mehr Zeit für die eigentliche Testung
- ▶ Verkürzung der Implementierungszeit



Noch Fragen?



Kundenumfrage zum Thema "digitale Validierungssoftware"



[Link zur Umfrage](#)



Dr. Susan Spiller
Validierungsingenieurin
Fachverantwortliche CSV

Tel.: +49 1512 9236404

E-Mail: sspiller@testotis.de

