

Wie wird validiert?

Systemvalidierung Schulung (Modul 3)

Version: V03

Ronald Speer

LIFE Leipzig

- ▶ validierungspflichtige Systeme
- ▶ Kategorisierung nach GAMP
- ▶ Rollen und Verantwortlichkeiten nach GAMP
- ▶ Das V-Modell
- ▶ Lebenszyklusmodell
- ▶ Angepasstes V-Modell
- ▶ Verantwortlichkeiten des Systemlieferanten
- ▶ Lieferanten-Audit
- ▶ Eigenentwicklungen
- ▶ Software Design Spezifikation
- ▶ Projektmanagement
- ▶ SOPs
- ▶ Deliverables gemäß SVMP
- ▶ Traceabilitymatrix

Validierungspflichtige Systeme

- ▶ Zur korrekten Fokussierung der Validierungsaktivitäten muss in einem Verbund zu jedem Zeitpunkt bekannt sein, welche Computersysteme validierungspflichtig sind und welche nicht
- ▶ **Validierungspflichtig sind grundsätzlich alle GCP-relevanten Computersysteme**, d.h. Computersysteme, die einen direkten oder indirekten Einfluss auf GCP-Daten haben
- ▶ Vorausgesetzt wird: eine genaue Kenntnis der Systeme, ihrer Komponenten und Schnittstellen
- ▶ Bewertung, nicht nur auf der Systemebene, sondern auch auf der Ebene der Komponenten und Schnittstellen ist notwendig
- ▶ Die Verbünde besitzen zur Unterstützung der sicheren Durchführung ihrer klinischen Forschungsprozesse eine Sicherheitsinfrastruktur

Kategorisierung nach GAMP

- ▶ Wegen der Komplexität der verschiedenen Computersysteme, die in klinischen Studien eingesetzt werden, hat ISPE / GAMP ein **5-stufiges Klassifikationsschema entwickelt**
- ▶ Es kann als Leitlinie dienen, **Validierungsanforderungen** der Komplexität des Systems zuzuordnen
- ▶ Jede Kategorie hat ihre eigenen Validierungsanforderungen, die von minimalen Anforderungen für Kategorie 1 und maximalen Anforderungen für Kategorie 5 reichen

Die einzelnen Kategorien (1)

Kategorie 1 (Betriebssysteme):

- ▶ das sind etablierte, kommerzielle Betriebssysteme. Beispiele sind Unix, Windows und VMS. Applikationen laufen unter der Kontrolle dieser BS. Etablierte BS sind nicht Subjekt einer spezifischen Validierung

Kategorie 2 (Firmware):

- ▶ Instrumente und Kontrolleinheiten beherbergen oft Firmware. Firmware ist die Kombination aus einem Hardware-Device, Computerinstruktionen und Daten, die als „Read only“ Daten im Instrument sich befinden
- ▶ Wird seit Gamp5 nicht mehr verwendet

Die einzelnen Kategorien (2)

Kategorie 3 (Standard Software Pakete):

- ▶ allgemein vorhandene Standard Software-Pakete, die eine „Off-the Shelf“ Lösung zu einem Prozess anbieten. Das Pakete ist nicht so konfiguriert, dass es die Prozesse selber definiert. Konfiguration ist limitiert auf die Etablierung der Run time Umgebung des Systems (z.B. Netzwerk)

Kategorie 4 (konfigurierbare Software-Pakete):

- ▶ Interfaces und Funktionen, die es erlauben, nutzerspezifische Prozesse zu konfigurieren (z.B. SAP). Dies umfasst auch die Konfigurierung definierter Softwaremodule.

Kategorie 5 (Custom bespoken Software):

- ▶ wird gemäß den Anforderungen des Nutzers speziell für ihn entwickelt.

Kategorien für Software und Validierung



Kategorie	Komponententyp	Validierungsansatz
1	Betriebssystem	IQ
3	Standardsoftware	
3a	Allgemeine Standardsoftware	IQ, OQ SOPs, Verwendungsbereich Schulung
3b	Spezielle Standardsoftware	IQ, OQ SOPs, Verwendungsbereich Schulung Herstellerbewertung
4	Konfigurierbare Softwarepakete	IQ, OQ SOPs, Verwendungsbereich Schulung Herstellerbewertung Jegliche kundenspezifische, individuelle Programmierung ist gemäß Kategorie 5 zu behandeln
5	Individualsoftware	DQ, IQ, OQ SOPs, Verwendungsbereich Schulung Hersteller-Audit

Besonderheit Standard-Softwarepakete

- ▶ Standard-Softwarepakete bieten eine „Off-the-shelf“-Lösung für bestimmte Geschäftsprozesse.
- ▶ Konfigurationsmöglichkeiten beschränken sich im Wesentlichen darauf, das System in seine technische Umgebung zu integrieren
- ▶ Die Kategorie der Standard-Softwarepakete ist weiter zu differenzieren in allgemeine Standardsoftware und spezielle Standardsoftware
- ▶ Aus diesem Grund wurde Kategorie 3 in die beiden Subkategorien 3a und 3b unterteilt:
 - ▶ Unter die Kategorie allgemeine Standardsoftware (3a) fallen z.B. Datenbankmanagementsysteme oder Textverarbeitungssysteme
 - ▶ Bei spezieller Standardsoftware (3b) handelt es sich um Branchensoftware, wie z.B. Datenmanagementsysteme oder EDC-Systeme für klinische Studien
- ▶ Die Validierungsansätze der beiden Softwareklassen unterscheiden sich in der Notwendigkeit einer Herstellerbewertung im Falle spezieller Standardsoftware.

Konfigurierbare Softwarepakete

- ▶ konfigurierbare Softwarepakete: weitgehende **Konfigurationsmöglichkeiten** nicht nur bei der Integration des Systems in seine technische Umgebung, diese Systeme können sehr genau an die einrichtungsspezifischen Geschäftsprozesse angepasst werden
- ▶ Konfigurationsmöglichkeiten reichen von der Deaktivierung nicht benötigter Funktionen über die Änderung benötigter Funktionen bis hin zur Programmierung zusätzlicher Module
- ▶ Der Validierungsansatz umfasst neben einer Herstellerbewertung insbesondere die Sicherstellung der Erfüllung der **Benutzeranforderungen** unter besonderer Berücksichtigung des konfigurierten Geschäftsprozesse
- ▶ Zusätzlich entwickelte bzw. stark veränderte Module fallen unter die Kategorie 5 und erfordern damit den maximalen Validierungsaufwand

Systemklassifikation



[1-011-anhang-systemklassifikation.xlsx](#)

- Die Person, die letztlich für die Führung des Geschäftsprozesses oder der Geschäftsprozesse verantwortlich ist.
- Diese Person ist normalerweise der Leiter der Funktionseinheit oder der Abteilung, die dieses System verwendet.
- Der Prozessinhaber muss verantwortlich sicherstellen, dass das computergestützte System und dessen Betrieb gemäß der anzuwendenden standardisierten Arbeitsanweisungen (SOPs) während seiner gesamten Einsatzdauer vorschriftenkonform und für den vorgesehenen Zweck geeignet sind.

Systeminhaber

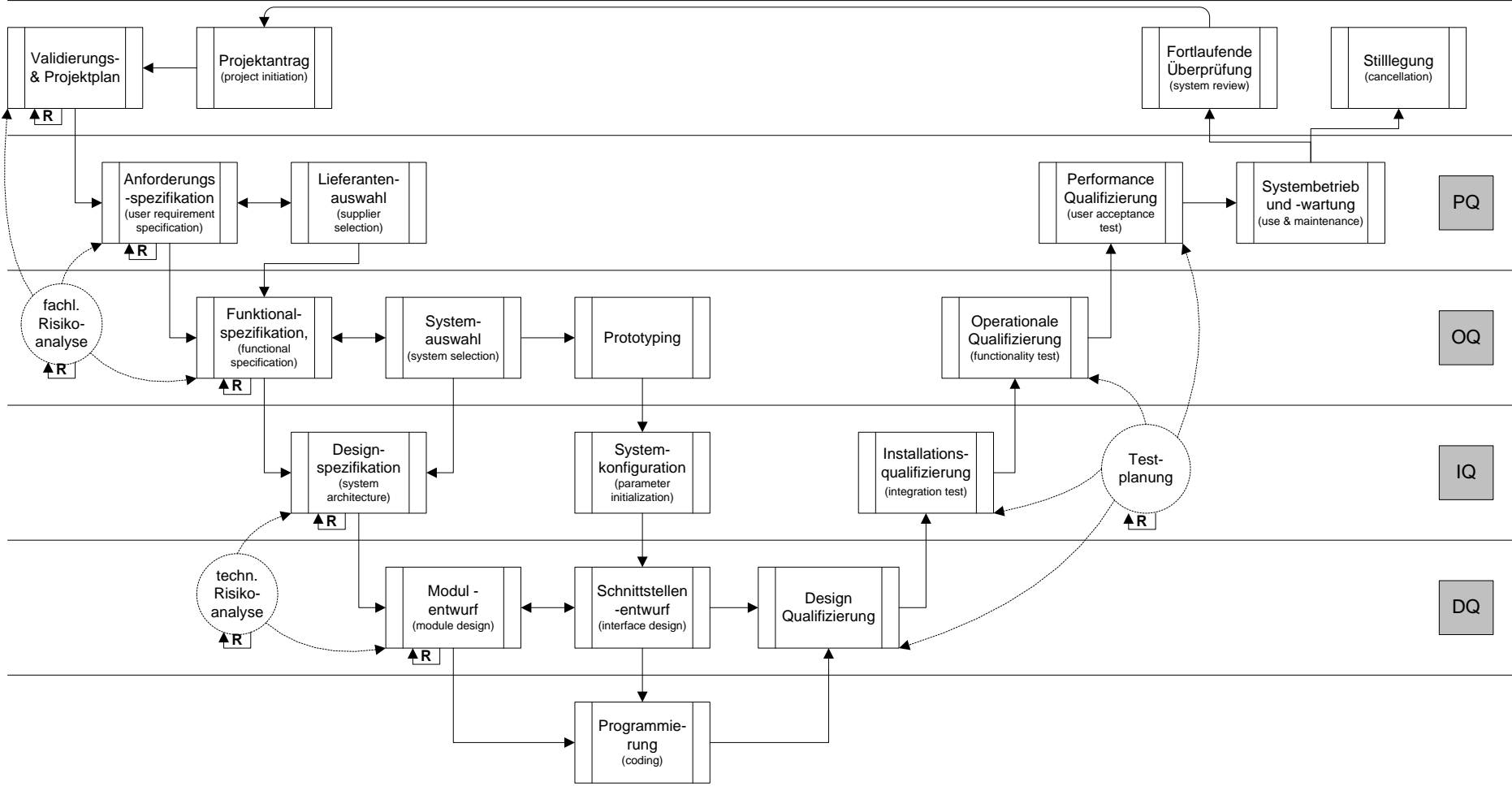
- Die Person, die letztlich für die Verfügbarkeit, den Service und Instandhaltung eines Systems und für die Sicherheit der auf diesem System gespeicherten Daten verantwortlich ist.
- Diese Person ist normalerweise der Leiter der Abteilung, die für den Systemservice und die Instandhaltung zuständig ist, obwohl die Rolle eigentlich auf den spezifischen Systemkenntnissen und nicht auf der Position innerhalb der Organisation basiert sein sollte.
- Der Systeminhaber ist dafür verantwortlich, sicherzustellen, dass das computergestützte System gemäß der SOPs unterstützt und gepflegt wird.

Die Verantwortlichkeit für die Kontrolle des Systemzugriffs sollte zwischen dem Prozessinhaber und dem Systeminhaber abgestimmt werden. In einigen Fällen kann der Systeminhaber auch der Prozessinhaber sein

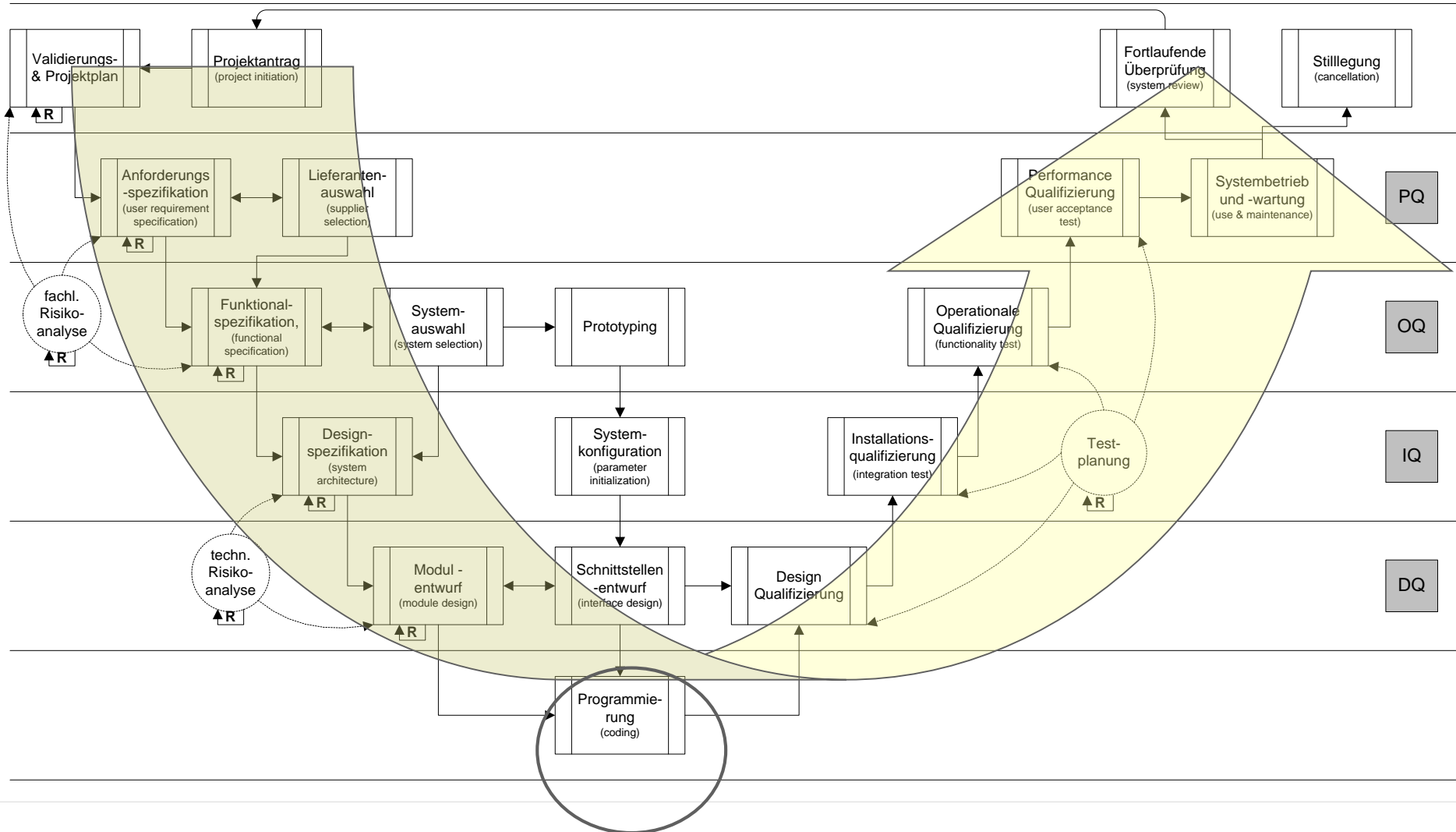
Fachexperte (FE)

- Jene Personen mit **spezifischer Fachkenntnis** in einem speziellen Bereich oder Feld. Fachexperten sollten die Führungsrolle bei der Verifizierung computergestützter Systeme innehaben.
- Die Fachexperten sind zuständig für die Planungs- und Verifizierungsstrategien, die Festlegung der Akzeptanzkriterien, die Wahl der geeigneten Testmethoden, die Durchführung der Verifizierungstests.

Das V-Modell



IT Lebenszyklusmodell



Warum das Lebenszyklusmodell (Validation Life Cycle)?

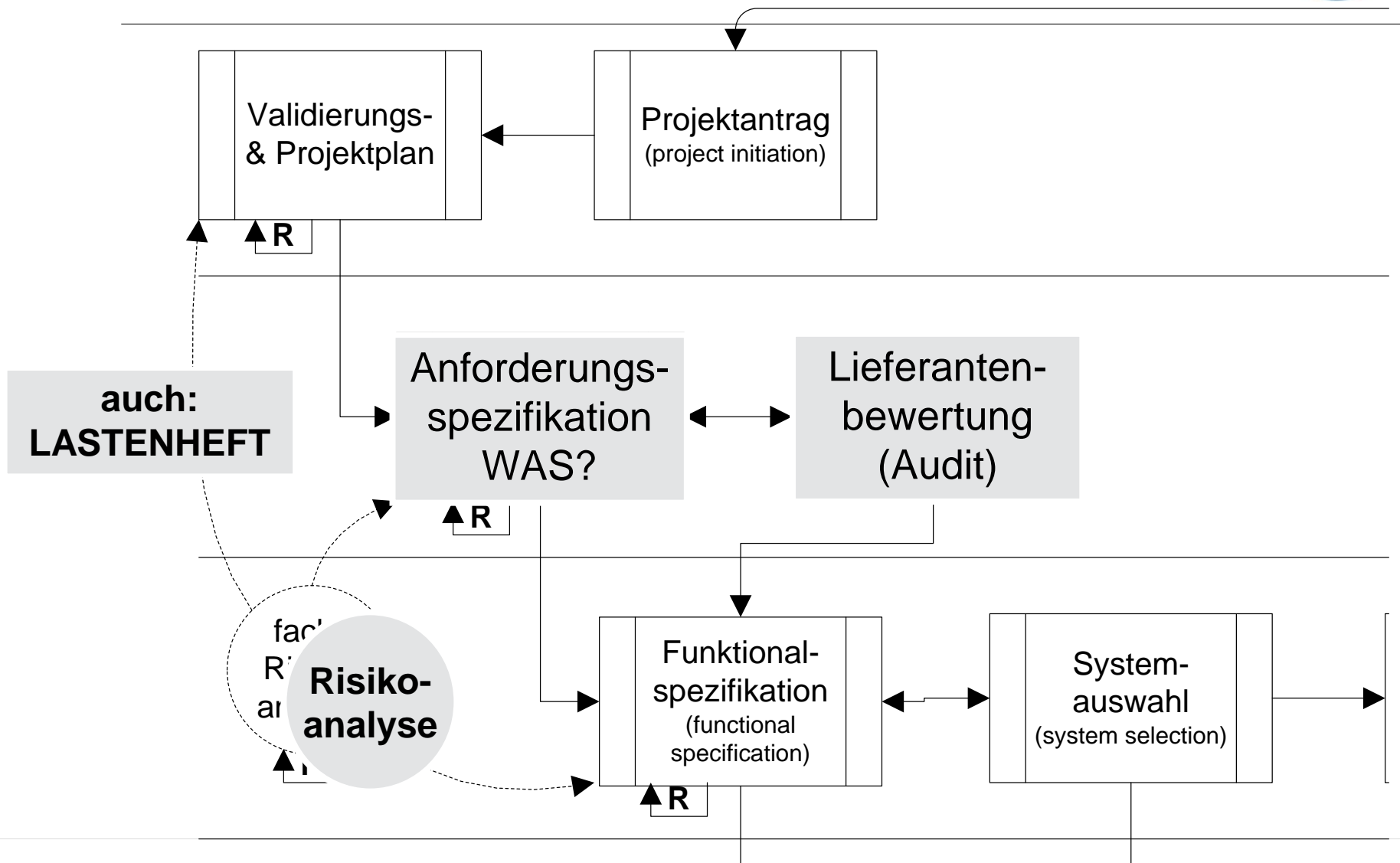
Die **Entwicklung, Implementierung und der Betrieb** von IT-Systemen, die regulativen Anforderungen entsprechen müssen, erfolgen anhand des IT Lebenszyklusmodells (VLC).

Dieses Modell gewährleistet:

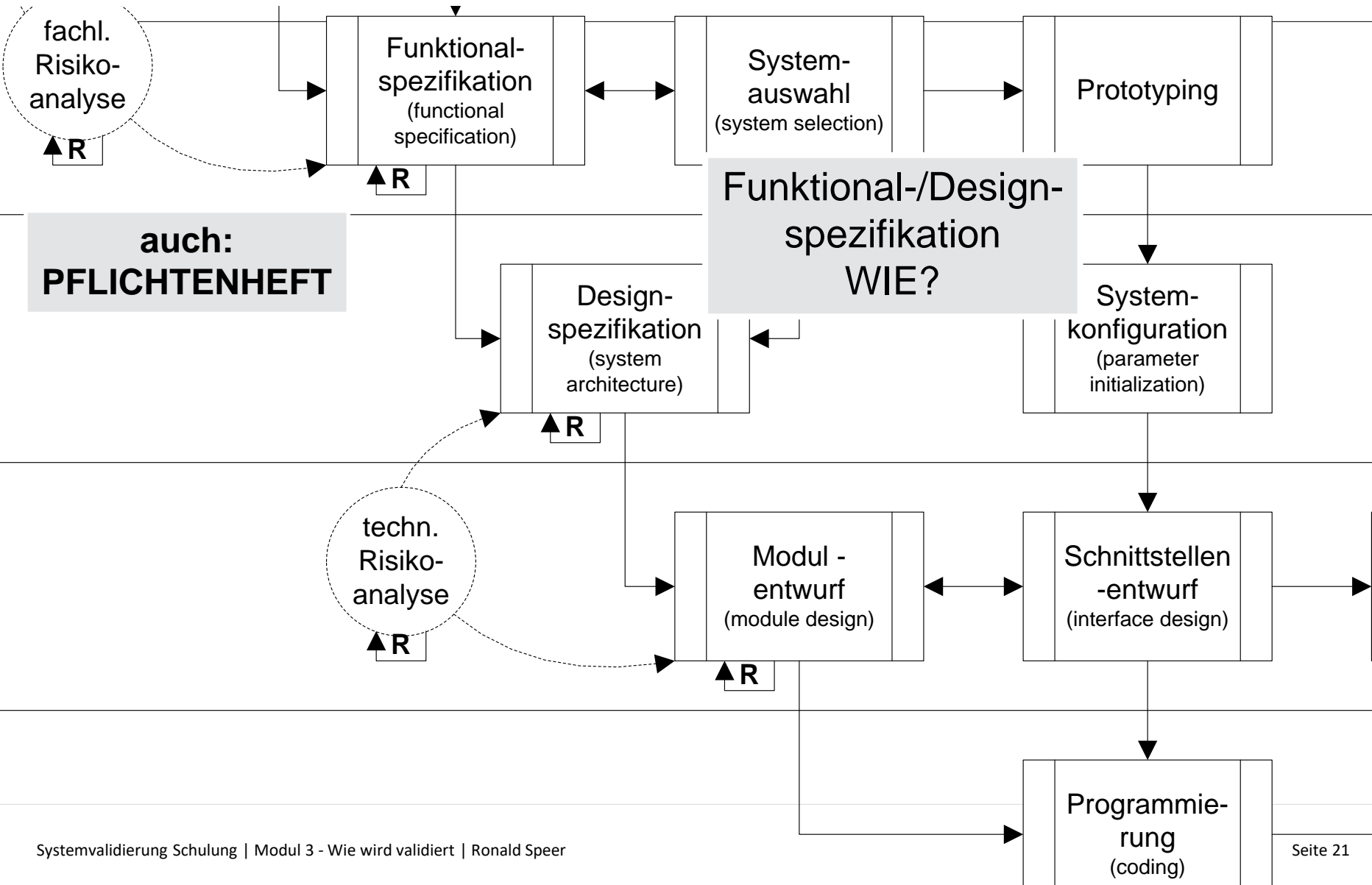
- ▶ dass das betreffende IT-System den Qualitätsanforderungen entspricht
- ▶ dass sämtliche Aktivitäten entsprechend den SOPs durchgeführt und dokumentiert werden
- ▶ dass die gesamte Systemdokumentation bei Änderungen - auch während der Entwicklung - konsistent bleibt.

(Der VLC entspricht dem Entwicklungs-Lebenszyklusmodell, dem sog. V-Modell, des GAMP-Leitfadens zur Validierung automatisierter Systeme in der pharmazeutischen Herstellung.)

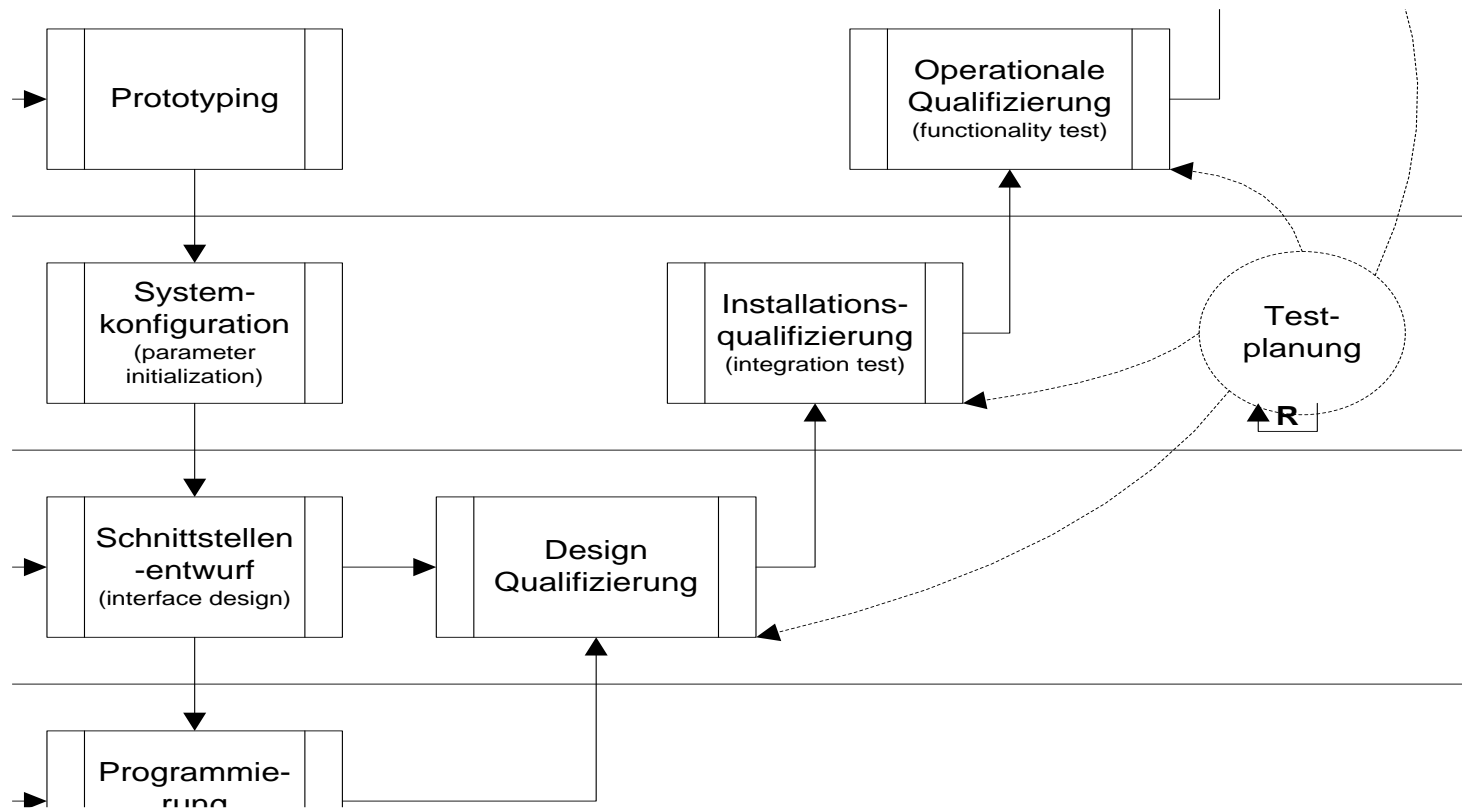
Phase I: Planung und Anforderungen



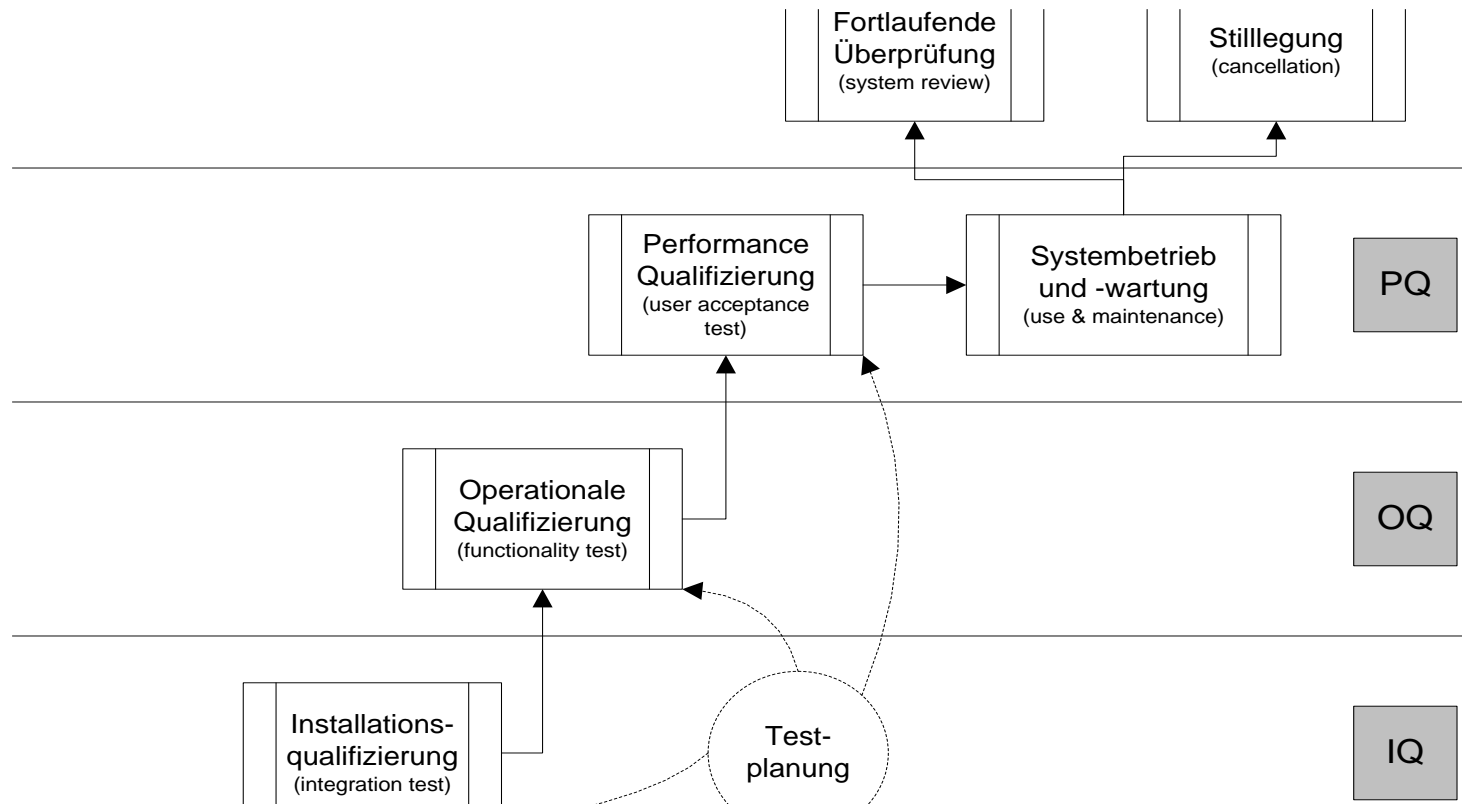
Phase II: Spezifizierung und Realisierung



Phase III: Planung und Durchführung von Tests



Phase IV: Akzeptanztest und Inbetriebnahme



Angepasstes V-Modell

- ▶ Vom klassischen V-Modell der Computervalidierung aus GAMP kann für bestimmte GCP-Systeme abgewichen werden
- ▶ Dabei können, basierend auf einer Risikobewertung, einzelne Phasen aus dem V-Modell zusammengelegt werden, z.B.:
 - ▶ Vereinigung der Anwenderspezifikation (AS) und Funktionale Spezifikation (FS) in der Traceability Matrix (= Liste der Nutzeranforderungen)
 - ▶ Kombination von IQ und OQ zur Systemqualifizierung durch den Softwareprovider
 - ▶ Zusammenfassung von OQ- und PQ-Tests
- ▶ Die Abweichungen vom V-Modell sind zu begründen und im Validierungsplan zu dokumentieren

Need-to-have



Alle Beteiligten müssen hinreichend **qualifiziert** sein
□ **Trainingsplan**; Vereinbarungen mit Externen

Die **Dokumentation** muss vollständig sein
□ Regelwerk (**QM-Handbuch, SOPs**)

Das System muss hinreichend **getestet** werden
□ **Testplan**

Die **Zugriffe** müssen geregelt sein
□ **Berechtigungskonzept**

Änderungen müssen kontrolliert werden
□ **Change Control Verfahren**

Das System muss die regulatorischen Anforderungen
funktional unterstützen
□ **Spezifikationen** (**Requirements Tracking**)

Der **Betrieb** des Systems muss geregelt werden
□ **Datensicherheitskonzept** (**Disaster Recovery**)

Zusätzlich zu beachten für Eigenentwicklungen

- ▶ Definition
 - ▶ Tool (z.B. SDTM-Wandler, SAS-Macros)
- ▶ Programmier-/Entwicklungsrichtlinien
- ▶ Modularisierung
- ▶ Fehlerbehandlung
- ▶ Code Review
- ▶ Detail-Spezifikation
- ▶ Testen auf Programm- und Modul-Ebene
- ▶ Change Management / Versionsverwaltung
- ▶ Qualifizierungsnachweise der Entwickler/Programmierer
- ▶ Programm-Dokumentation
- ▶ i.w.S. sind interne Entwickler wie externe Lieferanten zu bewerten (Qualitätsstandards, Nachvollziehbarkeit)

Code Review

Careful review of source code and its documentation is especially important for assuring that process specifications, conditions, sequencing, decision criteria, and formulas have been properly incorporated into the computer program.

(FDA Compliance Policy Guide 7132a. 15)

Source Code Review sollte umfassen:

- ▶ Review durch unabhängige Person, die nicht an der Programmierung des Tools beteiligt war
- ▶ Review soll sicherstellen, dass die Programmierung in Übereinstimmung mit den Design Spezifikationen erfolgte
- ▶ Codierungsstandards wurden eingehalten
- ▶ „Dead Code“ wurde entfernt
- ▶ Source Code Review-Bericht wird erstellt

Software Design Spezifikation

- ▶ definiert, mit welchen **Softwarekomponenten** die funktionale Spezifikation realisiert werden soll
- ▶ Die einzelnen Komponenten werden hinsichtlich ihrer Funktionalität, ihres modularen Aufbaus, ihrer zugrunde liegenden Datenstrukturen und ihrer Interaktionen detailliert beschrieben
- ▶ Die Software Design Spezifikation erfolgt im Anschluss an die Erstellung der funktionalen Spezifikation.

Projektmanagement: Teamzusammensetzung und Schulung (1)

- ▶ Entwicklung des **Personals**:
 - ▶ qualifiziertes Personal mit ausreichend Erfahrung
 - ▶ sollte fähig sein Planung, Einführung, Betrieb, Anwendungsberatung und regelmäßige Prüfung von Computersystemen durchzuführen
- ▶ Die **Qualifikation** des Personals wird gemäß Ausbildung, Schulung und Erfahrung im Umgang mit Computersystemen beurteilt
- ▶ Prinzip: Personal darf nur entsprechend seiner Ausbildung und seiner Kenntnisse beschäftigt werden
- ▶ Responsibility Split (Situation in den Verbänden)
- ▶ Der Verantwortungsbereich des einzelnen soll schriftlich festgelegt sein
- ▶ Qualität und Sicherheit sollten beim Betrieb durch das Personal von Computersystemen immer berücksichtigt werden

Projektmanagement: Teamzusammensetzung und Schulung (2)

- ▶ Für die Schulung des Personals, das mit Computersystemen zu tun hat, muss der **Prozesseigner** (häufig der Leiter des Verbundes) sorgen
- ▶ Er muss dem Personal die Inhalte der Computersysteme betreffenden Richtlinien und behördlichen Anforderungen nahe bringen
- ▶ Die Verantwortung für Schulung betrifft auch Systementwickler in den Verbänden, Wartungs- und Supportpersonal und Personal, dessen Tätigkeit die dokumentierte Funktionsfähigkeit des Computersystems beeinflussen könnte

Projektmanagement: Ressourcen

Durch Projektorganisation effektiv und ressourcenschonend validieren

- ▶ Projektteam definieren
- ▶ Organisationsdiagramm erstellen
- ▶ Schnittstelle Projektteam und Qualitätssicherungsfunktionen definieren
- ▶ Validierungsplan
- ▶ Projektergebnisse (Deliverables) definieren
- ▶ Projektmeilensteine definieren
- ▶ Projektstätigkeit festlegen
- ▶ Personal zuordnen
- ▶ Geplante Start- und Endtermine für jede Tätigkeit festlegen
- ▶ Tätigkeiten als Diagramm darstellen

- ▶ Die Validierung endet mit einem **Validierungsbericht**, der das Ergebnis der Validierung beschreibt und diskutiert, ob die Akzeptanzkriterien der Qualifizierung erfüllt wurden
- ▶ Er gibt an, ob ein System **freigegeben** werden kann

SOPs (Standard Operating Procedures)

- ▶ Zentrales Instrument für Validierung
- ▶ Für jede SOP ist anzugeben, wer für ihre Erstellung, Prüfung und Freigabe verantwortlich ist

- ▶ Betriebskonzept:
 - ▶ folgende Bereiche müssen abgedeckt werden
 - Datenerfassung und –verarbeitung
 - Systemwartung
 - Backup, Recovery, Notfallplan
 - Sicherheit
 - Änderungskontrolle

Folgende Dokumente sind gegebenenfalls im Rahmen der Validierung zu erstellen:

1. Anwenderspezifikation
2. Funktionale Spezifikation
3. Hardware Design Spezifikation
4. Software Design Spezifikation
5. Traceability-Matrix
6. Systemklassifikation
7. Berichte zu Herstellerbewertungen
8. Hardware-Installationsprotokoll(e)
9. Netzwerkdiagramm
10. Hardware-Liste
11. Software-Installationsprotokoll(e)
12. Software-Liste
13. Ausdrücke ggf. während der Installation automatisch erstellter LOG-Dateien
14. IQ-Herstellertestskripte (durchgeführt und unterzeichnet)
15. OQ-Testskript(e) (durchgeführt und unterzeichnet)
16. PQ-Testskript (durchgeführt und unterzeichnet)
17. Validierungsbericht

Auswahl je System und Festlegung im Validierungsplan

Traceabilitymatrix

- ▶ Die Traceabilitymatrix dient dem Qualitätsmanagement als **Überprüfung**, ob alle Nutzeranforderungen in den entsprechenden Phasen vollständig getestet wurden
- ▶ Die Traceabilitymatrix besitzt mindestens folgende Spalten:
 - ▶ Nr. (Nummer),
 - ▶ Anforderung, Funktion,
 - ▶ Ausführung der Funktion
 - ▶ Risiko
 - ▶ Datum
 - ▶ Tester
 - ▶ Signatur

Traceabilitymatrix



[1-020 vorlage_traceabilitymatrix.xlsx](#)

Vielen Dank für Ihre Aufmerksamkeit!

Mehr Information:

<http://www.tmf-ev.de>