Software-Qualitätshandbuch

Q-Kriterien der Software-Entwicklung

Mai 2017

V.0.6

Erstellt durch: Gerd Nanz (extern) ITSV GmbH

Dokumentverwaltung

Verteiler

|  |  |  |  |
| --- | --- | --- | --- |
| **Name (alphab.)** | **Träger** | **Abteilung** | **Ort** |
| TBD |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Dokumentenhistorie

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Datum** | **Ersteller** | **Beschreibung der Änderungen** |
| 0.1 | 14.12.2016 | Gerd Nanz | Neuerstellung auf Basis der Interviews |
| 0.2 | 08.02.2017 | Gerd Nanz | Ausarbeitung nach Review und Diskussion |
| 0.3 | 15.02.2017 | Gerd Nanu | Als Konzeptdokument zum Review |
| 0.4 | 07.03.2017 | Gerd Nanz | Überarbeitung nach Review des Konzepts, Ergänzung |
| 0.5 | 15.04.2017 | Gerd Nanz | Adaption nach Workshop, Vervollständigen |
| 0.6 | 13.05.2017 | Gerd Nanz | Adaption nach Workshop, Vervollständigen |
|  |  |  |  |

QS-Prüfkreis

|  |  |  |  |
| --- | --- | --- | --- |
| **Zur Prüfung vorgelegt am** | **Prüfkreis** | **Zum Review zurück am** | **Freigegeben am** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Inhalt**

[1 Einleitung 6](#_Toc482771262)

[1.1 Zweck des Dokuments 6](#_Toc482771263)

[1.2 Gültigkeit des Dokuments 6](#_Toc482771264)

[1.3 Aufbau des Dokuments 6](#_Toc482771265)

[1.4 Begriffsbestimmungen und Abkürzungen 7](#_Toc482771266)

[1.5 Zusammenhang mit anderen Dokumenten 8](#_Toc482771267)

[1.6 Weiterführende Literatur 8](#_Toc482771268)

[1.7 Typographische Konventionen 9](#_Toc482771269)

[2 Einführung 10](#_Toc482771270)

[2.1 Qualitätsbegriff 10](#_Toc482771271)

[2.2 Qualitätskriterien aus abstrakter (theoretischer) Sicht 11](#_Toc482771272)

[2.3 Allgemeines zu Qualitätsanforderungen 12](#_Toc482771273)

[2.4 Aufbau der EXCEL-Matrix 14](#_Toc482771274)

[2.5 Beispiele für Qualitätskriterien für Artefakte/Aktivitäten 14](#_Toc482771275)

[2.6 Generische Sicht des Life Cycle 16](#_Toc482771276)

[3 Qualitätsanforderungen an Artefakte/Aktivitäten 18](#_Toc482771277)

[3.1 Übergreifendes 18](#_Toc482771278)

[3.2 Anforderung definieren 37](#_Toc482771279)

[3.3 Anforderung umsetzen 49](#_Toc482771280)

[3.4 Testprozess 57](#_Toc482771281)

[3.5 Inbetriebnahme 77](#_Toc482771282)

[3.6 Betrieb 85](#_Toc482771283)

[3.7 Applikationsbetrieb managen 91](#_Toc482771284)

[3.8 Sunset 91](#_Toc482771285)

[3.9 Sonstiges Objekte 91](#_Toc482771286)

[4 Offene Punkte 92](#_Toc482771287)

[5 Anhang A: Checklisten und Protokolle 92](#_Toc482771288)

[5.1 Detaillierung ISO 25010 92](#_Toc482771289)

[5.2 Vorlage Release Notes 93](#_Toc482771290)

[5.3 Checkliste Release Notes 93](#_Toc482771291)

[5.4 Checkliste Anforderungsspezifikation 93](#_Toc482771292)

[5.5 Inhalte Anforderungsspezifikation 94](#_Toc482771293)

[5.6 Inhalte Umsetzungskonzept 95](#_Toc482771294)

[5.7 Inhalte Design Review des Software Designs 96](#_Toc482771295)

[6 Anhang B: Arten der Überprüfung 96](#_Toc482771296)

[6.1 Existenzprüfung (Sichtkontrolle) 96](#_Toc482771297)

[6.2 Formales Review 96](#_Toc482771298)

[6.3 Inhaltliches Review 97](#_Toc482771299)

[6.4 Integrationstest 97](#_Toc482771300)

[6.5 Überprüfung der Nachweise 97](#_Toc482771301)

[6.6 Test 97](#_Toc482771302)

[6.7 Lessons Learned 97](#_Toc482771303)

[6.8 Vergleich 97](#_Toc482771304)

[6.9 Ausbildungsnachweis 98](#_Toc482771305)

[7 Anhang C: Q-Kriterien für Artefakte/Aktivitäten 98](#_Toc482771306)

[7.1 Aktualität 98](#_Toc482771307)

[7.2 Berücksichtigung im Design 98](#_Toc482771308)

[7.3 Eignung und Umsetzbarkeit 98](#_Toc482771309)

[7.4 Existenz 98](#_Toc482771310)

[7.5 Existenz der Entscheidung 98](#_Toc482771311)

[7.6 Existenz Mindestinhalt 98](#_Toc482771312)

[7.7 Grad der Testabdeckung 98](#_Toc482771313)

[7.8 Inhaltliche Korrektheit 98](#_Toc482771314)

[7.9 Korrekte Funktionalität 98](#_Toc482771315)

[7.10 Nachweis der Einhaltung 99](#_Toc482771316)

[7.11 Nachweis der Nutzung 99](#_Toc482771317)

[7.12 Qualifikation des durchführenden Personals 99](#_Toc482771318)

[7.13 Qualitative Übereinstimmung 99](#_Toc482771319)

[7.14 Risikobasierte Berücksichtigung 99](#_Toc482771320)

[7.15 Übereinstimmung mit Anforderungen und Erwartungen 99](#_Toc482771321)

[7.16 Vollständige Berücksichtigung 99](#_Toc482771322)

[7.17 Vollständigkeit 99](#_Toc482771323)

[8 Anhang D: Referenzierte Rollen 99](#_Toc482771324)

# Einleitung

## Zweck des Dokuments

Das Dokument stellt Q-Kriterien für die Software- und Systementwicklung innerhalb der ITSV GmbH zu­sammen. Ziel ist es, den Qualitätsbegriff zu vereinheitlichen und die Kriterien für die Qualität von Soft­ware messbar zu machen.

Dazu wurden im 2. Halbjahr 2016 Interviews mit allen Beteiligen der Software-Entwicklung geführt und die bestehenden Q-Kriterien erhoben. Im 1. und 2. Quartal 2017 fanden einige Workshops und Ab­stim­mungs­­meetings statt. Dabei wurden sowohl agile Methoden als auch sequentielle Metho­den berück­sich­tigt.

Die Definition der Q-Kriterien wurde so angelegt, dass diese zu den Quality Gates im Software Entwicklungsprozess (siehe 1.5) zugeordnet werden können.

## Gültigkeit des Dokuments

Gültig für die gesamte Software-Entwicklung der ITSV GmbH.

## Aufbau des Dokuments

Das Dokument ist folgendermaßen aufgebaut:

*Kapitel 1* enthält die allgemeinen organisatorischen Informationen zum Dokument, um die Ein­bet­tung in die ITSV GmbH darzustellen.

*Kapitel 2* enthält eine Einführung zum Thema Q-Kriterien aus abstrakter und theoretischer Sicht so­wie einige in der Literatur und in internationalen Standards und Good Practices verwendete Qua­litätskriterien. Dieses Kapitel dient dem allgemeinen Verständnis, weshalb messbare und ver­ein­barte Q-Kriterien sinnvoll sind. Ebenso werden die Abhängigkeiten der einzelnen Aspekte wir Pro­zess, Q-Kriterium, Aktivität und Artefakt erklärt.

*Kapitel 3* ist das Kernkapitel des Dokuments und enthält die Q-Kriterien der ITSV GmbH, wie sie zu­nächst in den Interviews angesprochen – teilweise als Wünsche, teilweise als bereits beste­hend – und in der Folge zusammengeführt wurden. In diesem Kapitel werden die Q-Kriterien auf Artefakte, Themen und Aktivitäten bezogen. Jedes Artefakt/Aktivität/Thema wird in der folgenden Struktur präsentiert:

* Titel des Themas/Artefakts/der Aktivität
* Arten der Überprüfung des Q-Kriteriums
* Beschreibung des Themas/Artefakts/der Aktivität
* Beschreibung des Q-Kriteriums
* Einordnung nach ISO 25010
* Zweck und Nutzen des Themas/Artefakts/der Aktivität
* Bedingungen, wann das Q-Kriterium erfüllt ist[[1]](#footnote-1)
* Guidance
* Abhängigkeiten

Die Strukturierung erfolgt auf Basis der wesentlichen Aktivitäten und Artefakte im Sinne eines Soft­ware Life Cycle, anwendbar für (nahezu) alle Vorgehensmodelle in der Reihenfolge des Software Life Cycle. Dabei wird eine Aktivität/ein Artefakt üblicherweise in der Phase des Life Cycle angeführt, in der sie/es erstmals genutzt wird.

Die Einordnung nach ISO 25010 erfolgt, um eine Unterscheidung der Kriterien zwischen Produkt­kri­terium und Prozesskriterium zu erleichtern. Die Liste aus ISO 25010 wurde um den Eintrag „Sonstiges“ erweitert, in dem alle Prozess- und Dokumentationspunkte eingeordnet sind.

*Kapitel 3.9* enthält offene Punkte, die in der Folge bei der Verknüpfung dieses Dokuments mit dem Software-Entwicklungsprozess beachtet und geklärt werden müssen.

*Kapitel 5* enthält als Anhang A Hilfsmittel für die Umsetzung der in diesem Dokument beschriebenen Prüfmethoden (z. B. Checklisten, Protokolle, Referenzen auf Vorlagen).

*Kapitel 6* enthält als Anhang B eine Kurzbeschreibung der in diesem Dokument referenzierten Arten der Überprüfung.

*Kapitel 6* enthält als Anhang C eine Kurzbeschreibung der in diesem Dokument referenzierten Q-Kriterien für Artefakte/Aktivitäten.

*Kapitel 7.1* enthält als Anhand D eine Liste der referenzierten Rollen.

## Begriffsbestimmungen und Abkürzungen

|  |  |
| --- | --- |
| **Akronym** | **Erklärung** |
| Aktivität | Eine Tätigkeit, bei der ein oder mehrere Artefakte entstehen können, jedoch die Tätigkeit im Vordergrund steht. („Qualitätsobjekt“) |
| Artefakt | Ein Objekt, das aus einer Tätigkeit entsteht, z. B. Software Source Code, Software-Paket, Dokument, wobei aber das Objekt im Vordergrund steht. („Qualitätsobjekt“) |
| CAB | Change Advisory Board |
| CR | Change Request |
| DIN | Deutsche Industrie Norm |
| EN | Europäische Norm |
| EPIC | „Große“ User Story (die in mehrere User Stories geteilt werden kann) |
| ID | Identifikator |
| IEC | International Electrotechnical Commission |
| ISO | International Organization for Standardization |
| IT | Informationstechnologie |
| ITIL | Information Technology Infrastructure Library |
| LCM | Life Cycle Management |
| MS | Microsoft |
| PDF | Portable Document Format (Standard-Format für nicht veränderbare Dokumente) |
| Q-Kriterium | Qualitätskriterium |
| Quality Gate | Ein Quality Gate ist ein Meilenstein im Softwareentwicklungsprozess, zu dessen Erreichung definierte Q-Kriterien erfüllt sein müssen. Sind die Q-Kriterien nicht erfüllt, kann das Vorhaben nicht weitergeführt werden, bevor die Q-Kriterien nicht erfüllt wurden.  Zu einem Quality Gate gehören   * Ein Prüf-Arbeitsschritt, in dem die dem Quality Gate zugrundeliegenden Q-Kriterien geprüft werden. Diese Arbeiten können auch in potentiell mehreren, vor dem Quality Gate gelagerten Arbeitsschritten enthalten sein * Ein Entscheidungsschritt, in dem auf Grund der Prüf-Ergebnisse die Entscheidung für das weitere Vorgehen oder die Rückkehr zu einem zuvor gelagerten oder eigens dafür definierten Korrektur/Erstellungs-Arbeitsschritt getroffen wird. |
| QS | Qualitätssicherung |
| RDM | Release und Deployment Management |
| SAK | Softwarearchitekturkonzept |
| SLA | Service Level Agreement |
| UML | Unified Modeling Language |
| WIKI | Website, deren Inhalte von den Besuchern nicht nur gelesen, sondern auch direkt im Webbrowser geändert werden können |

## Zusammenhang mit anderen Dokumenten

* PzSWE Ausgabe 2.0 im Stand vom 23.04.2017
* Definition SAK V 1.7
* Spezifikation Autodeployment
* Beschreibung des Testprozesses

## Weiterführende Literatur

Folgende Normen und Standards wurden als Grundlage für die Definitionen und Inhalte der Qualitätsobjekte verwendet. Sie sind ind diesem Sinn nicht Vorgabedokument für die ITSV GmbH.

* DIN EN ISO 8402:1995-08, „Qualitätsmanagement – Begriffe“
* V-Modell XT 2.0
* ISO 250xx, Software engineering - Software product Quality Requirements and Evaluation (SQuaRE) - Guide to SQuaRE”
* ISO/IEC 12207:2008, „Systems and software engineering -- Software life cycle processes“

## Typographische Konventionen

Hervorhebungen werden *kursiv* geschrieben.

Beispiele sind mit einem Rahmen versehen.

# Einführung

## Qualitätsbegriff

### Qualität

Qualität wird in der Literatur und im Sprachgebrauch unterschiedlich definiert und verwendet. Eine häufig zu findende Interpretation ist:

„Qualität ist der Grad der Übereinstimmung zwischen Ansprüchen und Erwartungen (Soll) an ein Produkt und dessen Eigenschaften.“

Im gleichen Sinn definiert die DIN EN ISO 8402 Qualität als „die Gesamtheit von Merkmalen einer Einheit bezüglich ihrer Eignung, festgelegte und vorausgesetzte Erfordernisse zu erfüllen.“

In diesem Sinne ist Qualität nichts Absolutes, sondern ein wertneutraler Begriff, der erst durch Festlegung und Vereinbarung von Kriterien als gut und schlecht interpretiert werden kann.

Der Grad der erreichbaren Qualität hängt stark von den Rahmenbedingungen ab. Die verfügbare Zeit, die Qualifikation der beteiligten Personen, die Kundenanforderungen und -erwartungen beeinflussen diesen Grad ebenso wie das verfügbare Budget. Dabei muss aber darauf geachtet werden, dass in jeder Branche und Firmenkontext in jedem Fall Mindestanforderungen erfüllt sein müssen.

Für eine einheitliche Qualität der Software in der ITSV GmbH sind daher Vereinbarungen über die Kriterien und deren Grenzwerte erforderlich. Die Grundlagen für eine einheitliche Qualität werden in diesem Software-Qualitätshandbuch dokumentiert.

### Konstruktives Qualitätsmanagement

Das konstruktive Qualitätsmanagement will über die Festlegung von Vorgehensweisen und Methoden erreichen, dass das entstehende Softwareprodukt a priori bestimmte Eigenschaften besitzt. Hierzu verwendet man beispielsweise Methoden, Sprachen, Werkzeuge, Richtlinien, Checklisten und Standards bzw. deren zielführende Verwendung. Idealerweise sind diese Maßnahmen im Software-Entwicklungsprozess verankert, um so eine gleichbleibende Qualität zu erreichen.

### Analytisches Qualitätsmanagement

Das analytische Qualitätsmanagement umfasst das Testen von System-Bestandteilen und des Gesamtsystems (z. B. Unit-Test, Systemtest, Integrationstest, User-Acceptance-Test). Dabei wird darauf geachtet, dass die Erfüllung aller Kundenanforderungen durch die Tests abgedeckt und damit entschieden werden kann, ob eine Anforderung oder Erwartung erfüllt wird oder nicht. Auch diese Maßnahmen sollen im Software-Entwicklungsprozess verankert sein, haben aber andererseits eine prüfende Funktion, um über die Erfüllung von Anforderungen und Erwartungen an das Produkt entscheiden zu können.

Qualitätssicherung erfolgt im Allgemeinen im Rahmen des analytischen Qualitätsmanagements und umfasst die operative Durchführung der Maßnahmen

**Beispiel:**

Der Testrahmenplan ist ein Artefakt des analytischen Qualitäts*managements*. Im Testrahmenplan werden üblicherweise die geplanten Testarten sowie die Testziele und Testendekriterien beschrieben. Die Inhalte müssen mit den expliziten und impliziten Anforderungen und Erwartungen der Stakeholder abgeglichen sein.

Der Testplan definiert den zeitlichen Ablauf von operativen Test, ggf. unter Berücksichtigung zusätzlicher Informationen wie Testumgebung oder Personalplanung. Der Testplan ist ein Artefakt der Qualitätssicherung und muss inhaltlich zum Testrahmenplan passen.

## Qualitätskriterien aus abstrakter (theoretischer) Sicht

### Allgemeine Betrachtungen für dieses Dokument

Qualitätskriterien sind generell messbare Anforderungen an ein Artefakt, eine Aktivität oder eine Kombination von beiden. Die Qualitätskriterien können zu übergeordneten Gruppen zusammengefasst werden, müssen aber für jedes einzelne Artefakt, für jede Aktivität und jede Kombination von beidem spezifisch definiert werden („Zielwerte“). Die Normenreihe ISO 250xx unterscheidet zwischen

* Interner Software-Qualität,
* Externer Software-Qualität,
* Qualität in der Benutzung.

Ein geringfügig anderer Blickwinkel unterscheidet zwischen dem Qualitätsbegriff (Modell) für

* Software,
* Qualität in der Benutzung,
* Datenqualität.

Wesentlich für die Betrachtung sind die Abhängigkeiten zwischen

* Software-Entwicklungsprozess,
* Produkt,
* Artefakt,
* Daten,
* Installation und Konfiguration,
* Nutzung durch Betrieb,
* Nutzung durch Anwender.

Der Prozess liegt außerhalb der Betrachtung dieses Dokuments, allerdings definiert dieses Dokument Anforderungen (Existenz, Qualität) an Artefakte, die in den Prozessen entstehen. Dieses Dokument schlägt auch Prüfungsmethoden vor und gibt Hinweise, wie diese Prüfungsmethoden angewendet werden können und welche typischen Schwierigkeiten dabei auftreten.

Die Gesamtheit der Artefakte stellt das Produkt dar. Aus diesem Grund betrachtet dieses Dokument typische Aktivitäten und Artefakte, wie sie in einem Software-Entwicklungsprozess vorkommen. Diese Aktivitäten und Artefakte müssen bestimmte Eigenschaften haben, um dem Qualitätsanspruch der ITSV GmbH gerecht zu werden.

### Allgemeine Grundsätze

Qualitätskriterien können für alle Artefakte im Rahmen des Software-Entwicklungsprozesses definiert werden. Sie müssen jedenfalls auf die Anforderungen und Erwartungen der Kunden, auf die Anforderungen der Organisation sowie auf die Kritikalität der Artefakte ausgerichtet sein.

Üblicherweise werden Qualitätskriterien für Dokumente aus dem Software-Entwicklungsprozess, für einzelne Abschnitte (Aktivitäten) im Software-Entwicklungsprozess sowie für das Softwareprodukt/das System selbst definiert und vereinbart.

In diesem Software-Qualitätshandbuch werden Artefakte und Aktivitäten betrachtet. Artefakte sind Objekte, welche aus einer Aktivität hervorgehen und bei denen das Objekt im Mittelpunkt steht (mit seinem Ziel, dem Zweck und der enthaltenen Information). Diese Artefakte finden sich typischer Weise in den zugrundliegenden Standards und Good Practices wieder.

**Beispiel:**

Benutzerdokumentation ist ein Artefakt, das für (nahezu) jedes Software-Produkt erforderlich ist. Ziel ist, dass der Benutzer die notwendigen Informationen zur Bedienung des Produkts erhält und in die Lage versetzt wird, nachlesen zu können, wie die einzelnen Schritte bei der Bedienung sind. Dabei ist das Medium der Benutzerdokumentation unerheblich. Es kann dies Online-Help sein, Online-Dokumentation in Form eines (redaktionell betreuten) WIKI, ein Handbuch in Papierform oder als PDF-Datei.

Aktivitäten resultieren – abstrakt gesprochen – in einer Gruppe von Artefakten. Diese Artefakte stehen jedoch nicht im Mittelpunkt, dafür steht das Gesamtergebnis der Aktivität mit ihrem Ziel im Fokus.

**Beispiel:**

Der Unit Test ist eine Aktivität, die aus mehreren Artefakten besteht, z. B. Testfalldefinition, Testdaten, Testausführungsprotokoll, Abschlussprotokoll Unit Test. Der Fokus liegt auf dem Ziel, im Unit Test Fehler in der Software zu identifizieren und die üblichen Ziele eines White Box Tests (z. B. Wartbarkeit, Einhaltung der Coding Rules) zu überprüfen. Die Bewertung ist ergebniszentriert (Testabdeckung, Erreichung der Testendekriterien), nicht artefaktzentriert (z. B. Existenz und Korrektheit des Abschlussprotokolls Unit Test).

## Allgemeines zu Qualitätsanforderungen

### Nachweisbarkeit von Qualitätsanforderungen

Alle Prüfungen zu den Qualitätskriterien sind nur dann (offiziell) erfolgt, wenn sie dokumentiert wurden. Die Arten der Dokumentation zum Nachweis der Überprüfung und Bewertung werden in diesem Dokument nicht durchgängig behandelt.

### Abhängigkeiten

Die Beschreibung des Q-Kriteriums bezieht sich auf das Thema/Artefakt/die Aktivität. Sind mehrere Einträge für das Q-Kriterium angegeben (nummeriert), stellt die Reihenfolge ein Reifegradmodell für die Bewertung des Q-Kriteriums dar. Die Darstellung ist inkrementell.

**Beispiel:**

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit

Als erste Stufe wird die Existenz eines Objekts betrachtet. An dieser Stelle werden keine Aussagen über die Qualität des Objekts gemacht (siehe 6.1).

Als zweite Stufe wird formal überprüft, ob das Objekt den geforderten Mindestinhalt besitzt. Auch hier wird noch keine Aussage über die Qualität dieses Mindestinhalts gemacht (siehe 6.2).

Als dritte Stufe wird geprüft, ob das Objekt inhaltlich korrekt ist und seinen gewünschten Zweck erfüllt. Diese Stufe umfasst formale und inhaltliche Qualitätsanforderungen und benötigt die Einbettung des Themas/Artefakts/der Aktivität in den Gesamtkontext des Software-Entwicklung siehe (6.3).

Zu jeder Stufe des Q-Kriteriums gehört eine Methode, wie dieses Q-Kriterium überprüft werden kann. Meist gibt es mehrere mögliche Arten der Überprüfungen. Insbesondere Existenzprüfungen sind sehr häufig automatisierbar.

Zu jeder Stufe des Q-Kriteriums müssen die Bedingungen festgelegt sein, wann dieses Q-Krite­rium zufriedenstellend erfüllt ist. Während sich die unteren Stufen der Reife meistens auf eine einfache Bewertungsskala zurückführen lassen, erfordern die oberen Stufen im Allgemeinen Abstimmungsbedarf und vor allem auch ein gemeinsames Verständnis zwischen den Personen, welche die Kriterien anwenden.

Zu den Q-Kriterien eines Produkts tragen im Allgemeinen viele Einzelteile bei. Um sicherzustellen, dass diese Einzelteile zueinander passen und ausreichend für die geforderte Produktqualität sind, müssen die Q-Kriterien einerseits, die Aktivitäten und Artefakte anderseits aus unterschiedlichen Fragestellungen betrachtet werden:

* Welches Artefakt trägt zu einem Q-Kriterium bei?
* Besteht Vertrauen, dass das Q-Kriterium ausreichend erfüllt sind, wenn alle Artefakte den geforderten Beitrag liefern?
* Welche Aktivität trägt zu einem Q-Kriterium bei?
* Besteht Vertrauen, dass das Q-Kriterium ausreichend erfüllt sind, wenn alle Aktivitäten in der geforderten Qualität ausgeführt werden?

Diese Fragestellungen erfordern die Möglichkeit, die Abhängigkeiten aus den unterschiedlichen Sichten auswerten zu können. Zu diesem Zweck liegt diesem Software-Qualitätshandbuch ein EXCEL-Matrix bei, in der durch Filterung die einzelnen Sichten in der Darstellung unterstützt werden.

## Aufbau der EXCEL-Matrix

In der Spalte A stehen die Artefakte und Aktivitäten, welche in der ITSV angesprochen wurden bzw. jedenfalls erforderlich sind.

Die Zeile 1 der EXCEL Matrix besteht derzeit aus vier verschiedenen Sichten und ist folgendermaßen aufgebaut:

### Sicht 1: Priorität

Die Priorität besteht aus 3 Stufen:

1. sehr wichtig
2. sinnvoll
3. weiterer Vorschlag

In der Spalte B ist ein Vorschlag seitens corda eingetragen.

In der Spalte C wird die Entscheidung der ITSV GMBH für die Priorität eingetragen.

### Sicht 2: In der ITSV erreichtes Level

In der Spalte E ist der Iststand erreicht, bezogen auf die in Kapitel 3 eingetragenen Qualitätskriterien. Diese werden mit den Buchstaben A … Z bezeichnet und sind inkrementell. A ist die geringste Stufe, die erreicht werden kann, die weiteren Stufen bezeichnen die „Level“, die erreicht werden können. Dabei ist nicht notwendigerweise der höchste Level der für die ITSV sinnvolle anzustrebende Level. Die angeführten Level ergeben sich aus der konsequenten Einbettung der Artefakte/Aktivitäten in einen Referenzprozess nach ISO 12207.

In Spalte F ist ein Vorschlag seitens corda für den von der ITSV GmbH mindestens zu erreichenden Level eingetragen.

In der Spalte G wird die Entscheidung der ITSV GMBH für das angestrebte Level eingetragen.

### Sicht 3: Q-Kriterien der Artefakte/Aktivitäten

In den Spalten I bis AC werden die zu den einzelnen Artefakten und Aktivitäten gehörigen vorgeschlagenen Q-Kriterien angeführt. Die Buchstaben beziehen sich dabei auf das Level aus 2.4.2. Eine Kurzbeschreibung der Methoden ist in 6 zu finden.

### Sicht 4: Einordnung nach ISO 25010

In den Spalten AE bis AM werden die Beiträge des Artefakts bzw. der Aktivität zugeordnet. Ziel muss sein, dass in jeder Spalte der Q-Kriterien nach ISO 25010 mindestens eine Zuordnung ist, da sonst das Q-Kriterium nicht berücksichtigt ist.

### Benutzung der EXCEL Matrix

Die EXCEL Matrix erlaubt Suchen und Filtern, insbesondere auch Auswertungen in Bezug auf Abdeckung, Vollständigkeit und Konsistenz.

## Beispiele für Qualitätskriterien für Artefakte/Aktivitäten

Dieser Absatz dient der Illustration der Konzeption von Q-Kriterien und stellt beispielhaft mögliche Q-Kriterien vor.

### Anforderungsdokumentation

Die folgende Liste stellt Beispiele von Qualitätskriterien an Anforderungsdokumentation vor:

* Lesbarkeit
* Inhaltliche Korrektheit
* Formale Korrektheit
* Verständlichkeit
* Vollständigkeit
* Konsistenz
* Terminologie
* Testbarkeit

### Softwareprodukt/System

Die folgende Liste stellt Beispiele von Qualitätskriterien an ein Softwareprodukt/System vor:

* Funktionale Eignung (ISO 25010)
* Effizienz (ISO 25010)
* Kompatibilität (ISO 25010)
* Benutzbarkeit (ISO 25010)
* Zuverlässigkeit (ISO 25010)
* Sicherheit (ISO 25010)
* Wartbarkeit (ISO 25010)
* Portierbarkeit (ISO 25010)
* Abrechenbarkeit
* Nachweisbarkeit der Leistungsfähigkeit
* Revisionssicherheit

### Benutzerdokumentation

Die folgende Liste stellt Beispiele von Qualitätskriterien an Benutzerdokumentation vor:

* Zielgruppenorientierung
* Inhaltliche Korrektheit und Übereinstimmung mit System
* Verständlichkeit
* Formale Korrektheit
* Konsistenz
* Terminologie
* Vollständigkeit

### Betreibbarkeit (Übergabe an Betrieb)

Die folgende Liste stellt Beispiele von Themen und Qualitätskriterien für die Übergabe eines Softwareprodukts/Systems an den Betrieb vor:

* Qualifikation des Betriebs (Skills)
* Verfügbarkeit der Informationen
* Aktualität des Standes (Baselining)
* Klarheit der Verantwortungen

### Prozess

Die folgende Liste stellt Beispiele von Themen und Qualitätskriterien im Software-Entwicklungsprozess vor:

* Konsens über Anforderungen bei allen Stakeholdern
* Planungsqualität für die Umsetzung
* Aktualität der Anforderungen im Software Life Cycle (Transparenz bei Änderungen (Change Requests))

### Berichtswesen

Die folgende Liste stellt Beispiele von Themen und Qualitätskriterien für das Reporting von Entwicklungsprojekten vor:

* Fortschritt
* Aufwand
* Qualität
* Berichte an die Projektleitung
* Berichte an den internen Auftraggeber

## Generische Sicht des Life Cycle

Für die Gliederung der einzelnen Artefakte und Aktivitäten wird eine generische Sicht des Software Life Cycle verwendet. Diese sieht folgende Einteilung vor:

* Anforderung definieren
  + Vorbereitung
  + Konzept
  + Angebot
  + Auftrag
* Anforderung umsetzen
  + Analyse
  + Grobentwurf
  + Arbeitsplanung
  + Feinentwurf
  + Umsetzung
  + Test
  + Releasefähigkeit
  + Freigabe
  + Abnahme
  + Abschluss
* Inbetriebnahme
  + Produktivsetzung
* Betrieb
  + Monitoring
* Applikationsbetrieb managen
  + Applikationswartung
  + Fehleranalyse
  + Fehlerbehebung
* Sunset
  + Außerbetriebnahme

# Qualitätsanforderungen an Artefakte/Aktivitäten

## Übergreifendes

### Formaler Auftrag

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit dem Formalen Auftrag werden der durchführungsverantwortlichen Person die Entscheidungs­befugnis und die Verantwortung für die ordnungsgemäße Umsetzung der im Auftrag vereinbarten Ziele übertragen. Die beauftragende Person übernimmt damit die Managementposition und wird am Ende des Vorhabens die durchführungsverantwortliche Person bei Erfüllung der Vorgaben ent­lasten.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Artefakt Formaler Auftrag definiert im Sinne des Projektmanagements den Umfang (Scope), die Ziele und den Zweck des Vorhabens. Inhalte sind

* Ziele
* Nicht-Ziele
* Scope (Abgrenzung)
* Abhängigkeiten von anderen Vorhaben
* High Level Anforderungen
* Nutzen
* High Level Abnahmekriterien
* Grobschätzung des Aufwands
* Geplante Durchlaufzeit
* Kostenrahmen

Zielgruppen sind

* Interner Auftraggeber (Management)
* Produktmanagement
* Product Owner
* Alle Teilnehmer an der Umsetzung
* Betrieb
* Kunde
* Controlling

Der formale Auftrag muss in ausdruckbarer Form (z. B. PDF) vorliegen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer revisionssicheren Form, kann automatisiert werden.

Existenz Mindestinhalt: Review der ausgefüllten Struktur Vorgaben, kann automatisiert werden.

Inhaltliche Korrektheit: Review mit den Kriterien   
- Vollständigkeit des Auftrags  
- Abgrenzung des Auftrags  
- Verständlichkeit des Auftrags  
- Messbare Definition der Ziele im Auftrag  
- Plausibilität der Durchlaufzeiten  
- Plausibilität des Kostenrahmens

#### Guidance

Der formale Auftrag stellt im Allgemeinen den Vertrag zwischen internem Auftraggeber und Umset­zungs­verantwortlichen (z. B. Projektleiter) dar. Auf Basis dieses Auftrags werden die Rand­be­din­gungen des Vorhabens definiert. Insbesondere sollen im Auftrag auch die Erwartungen und An­for­de­rungen des internen Auftraggebers abgebildet sein, um dem Umsetzungs­ver­ant­wort­li­chen den not­wendigen Handlungsspielraum zu definieren.

Es ist ausreichend und sogar wünschenswert, wenn die angeführten Informationen in anderen Artefakten enthalten sind und auf sie referenziert wird, z.B.

* High Level Anforderungen in der Anforderungsspezifikation (siehe 3.2.1)
* High Level Abnahmekriterien in der Anforderungsspezifikation (siehe 3.2.1 und **Fehler! Verweisquelle konnte nicht gefunden werden.**)
* Grobschätzung des Aufwands (siehe 3.2.1 und **Fehler! Verweisquelle konnte nicht gefunden werden.**)
* Geplante Durchlaufzeit (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**)
* Kostenrahmen (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**)

#### Abhängigkeiten

* Anforderung des Kunden oder eines Gremiums
* Change Request (siehe 3.2.3)

### Planung der Entwicklung (Projektplan)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit einer konsequenten, realistischen und ausreichend detaillierten Planung auf Basis der Ziele und der Anforderungsspezifikation können Risiken im Entwicklungsablauf reduziert werden. Insbe­son­dere wird die Einbindung der einzelnen Stakeholder damit verbindlich vorher dokumentiert und ver­einbart.

#### Beschreibung des Themas/Artefakts/der Aktivität

Die Planung der Entwicklung (Projektplanung) ist ein Element des konstruktiven Qualitäts­mana­gements. Die Aktivität muss den allgemeinen Regeln des Requirements Engineerings und des Projekt­managements unter Berücksichtigung der Qualitätsanforderungen genügen.

Zielgruppen sind

* Interner Auftraggeber (Management)
* Projektteam
* Multiprojektmanagement
* Produktmanagement

Die Planung der Entwicklung kann in folgenden Formaten vorliegen:

* Ausdruckbares Dokument (z. B. PDF)
* Elektronisch in einem alleinstehenden Tool (z. B. Microsoft Project)
* Elektronisch in einem übergreifenden Tool (z. B. JIRA, Polarion)
* Auf Papier auf einem Wallpaper (z. B. Post-It an der Wand)

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Eignung und Umsetzbarkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer zu definierten Zeitpunkt unveränderbaren Form, die von der Zielgruppe gelesen werden kann; kann automatisiert werden

Existenz Mindestinhalt: Review der Planungstiefe gemäß Vorgaben

Eignung: Review der Projektplanung mit den Kriterien  
- Sind alle erforderlichen Punkte geplant?  
- Sind die geplanten Aufwände und Zeiten realistisch?  
- Stehen die geplanten Ressourcen zur Verfügung?  
- Sind die geplanten Ressourcen für die Aufgabe ausreichend qualifiziert?

#### Guidance

Planung der Entwicklung basiert idealerweise auf der Anforderungsspezifikation, indem die Lieferobjekte, die am Ende des Vorhabens vorhanden sein müssen, explizit deklariert und beschreiben werden. Hier gehören insbesondere auch die Qualitätskriterien für die Lieferobjekte (Abnahmekriterien) dazu. Dies Abhängigkeit von der Anforderungsspezifikation und ggf. sogar vom Umsetzungskonzept erfordert ein iteratives Vorgehen und kontinuierliche Überprüfung der Planung. Weiters sollten bei der Planung bereits Überlegungen für Abweichungen angestellt werden, insbesondere wenn Ressourcen wider Erwarten nicht zur Verfügung stehen oder ausfallen (Risikomanagement).

Es ist möglich, dass die Planung sukzessive mit der Verfeinerung der Anforderungen entsteht und erst zu Beginn der Umsetzung der Anforderungen (siehe 3.3) vollständig ist.

#### Abhängigkeiten

* Formaler Auftrag (siehe 3.1.1)
* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**)

### Abhängigkeiten von Produkten (IT-Map)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die IT-Map dient dem Überblick über alle oder zumindest die wesentlichen Produkte und ihre Abhängigkeiten auf hoher Ebene. Mit der IT-Map können die Zusammenhänge bei übergreifenden Änderungen und bei Neuentwicklungen leichter geplant und analysiert werden. Weiters dient sie einer „Inventur“ aller Produkte.

#### Beschreibung des Themas/Artefakts/der Aktivität

Die IT-Map ist ein Überblicksdokument, welches die Produkte oder die Komponenten der IT-Gesamtlandschaft darstellt.

Zielgruppen sind

* Interner Auftraggeber (Management)
* Betrieb
* IT-Architektur
* Testteam

Die IT-Map liegt im Allgemeinen in graphischer Form oder in Listenform vor. Häufig verwendete Tools für die Darstellung sind VISIO, EXCEL oder – wartungsfreundlicher – UML-Tools zur Darstellung von Kontextdiagrammen und Architekturverwaltungstools (z. B. ADO-IT), IT Asset Management Software usw.

Das Abhängigkeitsdiagramm (siehe 3.1.4) detailliert die IT-Map auf Schnittstellenebene.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Vollständigkeit
3. Inhaltliche Korrektheit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit | x |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der IT-Map in revisionssicherer Form, kann automatisiert werden

Vollständigkeit: Review mit den Kriterien  
- Sind alle Produkte aufgenommen?  
- ggf. Prüfung gegen die Aufzeichnungen des Configuration Managements

Inhaltliche Korrektheit: Review mit den Kriterien  
- Korrektheit der Abhängigkeiten  
- Beschreibung der wesentlichen Produkte  
- Erforderliche Beschreibungstiefe

*Tailoring*: Die Verpflichtung zum Eintrag eines Produkts in die IT-Map mit den wesentlichen Abhängigkeiten unterliegt nicht der Möglichkeit des Tailorings.

#### Guidance

Die IT-Map sollte sämtliche Produkte benennen und die wesentlichen Abhängigkeiten darstellen. Insbesondere wird an dieser Stelle der Bezug zu den Geschäftsprozessen hergestellt. Bei größeren Änderungen (produktübergreifend) sollte die IT-Map genutzt werden, um die Planung der Änderungen zu unterstützen und die Impactanalyse einfacher durchführen zu können.

Allein die Benennung aller Produkte auch ohne Darstellung der Abhängigkeiten bringt Nutzen, da hiermit eine „Inventarliste“ zur Verfügung steht, die als Checkliste für die Vollständigkeit der Analyse verwendet werden kann.

Die Prüfung der inhaltlichen Korrektheit und der Vollständigkeit der IT-Map kann – wenn nicht eine gleichartige und von allen genutzte Struktur der erforderlichen Informationen eingehalten wird – sehr aufwändig werden.

#### Abhängigkeiten

Abhängigkeitsdiagramm

Schnittstellendokumentation

SAK

### Abhängigkeiten von Produkten (Abhängigkeitsdiagramm)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Ein Abhängigkeitsdiagramm dient in der ITSV der Darstellung von Abhängigkeiten zwischen Applikationen. Damit wird einerseits die Fehlerbehebung erleichtert, andererseits wird die Impactanalyse bei Änderungen vereinfacht. Zusätzlich unterstützt das Abhängigkeitsdiagramm den Betrieb und das Applikationsmanagement bei der Installation der korrekten und erforderlichen Pakete.

#### Beschreibung des Themas/Artefakts/der Aktivität

Ein Abhängigkeitsdiagramm ist eine Darstellung von Abhängigkeiten zwischen abgegrenzten Objekten (z. B. Applikationen, Komponenten). In den Abhängigkeiten können unterschiedliche Beschreibungstiefen verwendet werden.

* Existenz einer Abhängigkeit
* Art der Abhängigkeit
* Details zur Abhängigkeit

Zielgruppen sind

* IT-Architektur
* Entwicklung
* Integrationsteam
* Testteam

Abhängigkeitsdiagramme liegen im Allgemeinen in graphischer Form oder in Matrizenform vor. Häufig verwendete Tools für die Darstellung sind VISIO, EXCEL oder – wartungsfreundlicher – UML-Tools zur Darstellung von Kontextdiagrammen und Architekturverwaltungstools (z. B. ADO-IT), IT Asset Management Software usw.

Es ist eine Detaillierung der IT-Map (siehe 3.1.1).

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit | x |
| Portierbarkeit | x |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.), falls nicht formal gegen die Entwicklungsdokumentation geprüft wird

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer unveränderbaren Form, die von der Zielgruppe gelesen werden kann, kann automatisiert werden

Vollständigkeit: Review der Abhängigkeitsmatrix mit den Kriterien  
- Sind alle Applikationen enthalten?  
- Korrektheit der eingetragenen Abhängigkeiten

#### Guidance

Die Existenz eines Abhängigkeitsdiagramms (üblicherweise in Form einer Matrix) mit den wichtigsten Applikationen unterstützt in den meisten Fällen bei Planung, Tests, Fehlersuche und auf organisatorischer Ebene. Allerdings bringt erst ein vollständiges Abhängigkeitsdiagramm *aller* Applikationen den weitreichenden Nutzen.

Zusätzlich können in einem Abhängigkeitsdiagramm die Arten der Abhängigkeiten eingetragen werden, z. B. welche Daten über eine Schnittstelle ausgetauscht werden, welche Seite der Schnittstelle die Plausibilität/Korrektheit der gesendeten/empfangenen Daten überprüft.

Fehler im Abhängigkeitsdiagramm sind bei deklarierter Vollständigkeit gefährlich, da sich die Nutzer im Allgemeinen auf die Information verlassen. Daher ist bei Änderungen besonders auf die Korrektheit zu achten.

Es ist möglich, dass das Abhängigkeitsdiagramm sukzessive mit der Verfeinerung der Anforderungen entsteht und erst im Laufe der Umsetzung der Anforderungen (siehe 3.3) vollständig ist.

#### Abhängigkeiten

* IT-Map (siehe 3.1.3)
* Systemarchitekturkonzept (siehe 3.2.5)
* Schnittstellendokumentation (siehe 3.3.1)

### Release Schein

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der Nutzen des Release Scheines ist die möglichst durchgängige Standardisierung der standard­mäßig verwendeten Software, um so den Wartungsaufwand im Rechenzentrum vertretbar zu hal­ten. Mangelnde Übereinstimmung mit dem Release Schein erhöht die Fehlerwahr­schein­lich­keit und erhöht üblicherweise die Wartungs-, Installations- und Betriebskosten.

#### Beschreibung des Themas/Artefakts/der Aktivität

Der Release Schein definiert die Systemumgebung, in welcher eine Applikation eingesetzt werden darf. Er umfasst die die zugelassenen Stände des Betriebssystems, von Middleware und standard­mäßig verwendeten Applikationen, auf die zugegriffen wird. Üblicherweise werden zwei unter­schied­liche Stände der Software im Release Schein angeführt. Benötigt eine Applikation andere Stän­de der Software, so muss dies explizit genehmigt werden.

Ein Release Schein soll nur verfügbare und vom Hersteller unterstützte Software enthalten.

Zielgruppen sind

* Entwicklung
* Betrieb
* Testteam
* Applikationsmanagement
* Produktmanagement

Der Release Schein ist entweder als ausdruckbares Dokument (z. B. PDF) oder in elektronischer Form (z. B. WIKI-Seite[[2]](#footnote-2)) verfügbar.

#### Beschreibung des Q-Kriteriums

1. Existenz der Entscheidung
2. Aktualität
3. Berücksichtigung im Design
4. Nachweis der Einhaltung
5. Existenz der Entscheidung (bzgl. Upgrade-Vorgehen)

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)
* Inhaltliches Review von Vorhabensdokumenten (für C.)
* Test (für D.)
* Existenzprüfung (Sichtkontrolle) (für E.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Release Scheines, kann automatisiert werden

Aktualität: Review mit den Kriterien  
- Sind alle Basiskomponenten im Releases Schein enthalten?  
- Sind die erlaubten Versionen der Basiskomponenten noch vom Hersteller unterstützt und besteht ein Wartungsvertrag (soweit anwendbar)?

Berücksichtigung: Review der Vorhabensdokumente mit dem Kriterium  
- Ist der Release Schein in der aktuellen Version berücksichtigt.

Einhaltung: Überprüfung beim finalen Test des Vorhabens

Upgradevorgehen: Entscheidung über das Vorgehen für jedes Produkt in Bezug auf Upgrade

*Tailoring*: Für die Nichteinhaltung des Release Scheines in einem Vorhaben ist eine dokumentierte Begründung erforderlich, die von TBD genehmigt werden muss.

#### Guidance

Genaugenommen ist der Release Schein kein Qualitätsobjekt, sondern eine Vorgabe an die Entwicklung. Er wurde trotzdem in dieser Liste aufgenommen, da er zentrale Bedeutung für Entscheidungen hat.

Es ist sinnvoll, die Upgrades auf die aktuellen Software Stände des Release Scheines im Prozess zu verankern und beim Projektstart explizit zu überprüfen (derzeit im Systemarchitekturkonzept (siehe 3.2.5)). Die Verankerung im Prozess passiert idealerweise im Software Life Cycle-Management, wobei Applikationsmanagement und Produktmanagement geeignete Zeitpunkt für Upgrades identifizieren müssen.

Der Release Schein hat zwei Aspekte, einerseits der Release Schein an sich, der die aktuellen von den Herstellern unterstützen Basiskomponenten enthalten sollte und bei Änderungen für alle Produkte entscheiden sollte, wie die weitere Vorgehensweise in Bezug auf Upgrades sein soll. Diese Entscheidung kann auch in regelmäßigen Abständen in einem gesonderten Review für die einzelnen Produkte getroffen werden.  
Andererseits muss der Release Schein angewendet werden, d. h. die einzelnen Vorhaben müssen in berücksichtigen und einhalten. Dies wird im SAK geplant und im finalen Test (spätestens) überprüft.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**)
* Upgradepfade (siehe 3.1.6)
* Systemarchitekturkonzept (siehe 3.2.5)

### Upgradepfade

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit der Festlegung der Upgradepfade werden Versionswechsel erleichtert und von Risiko befreit, wenn einzelne Zwischenversionen nicht produktiv gesetzt wurden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Thema Upgradepfade behandelt die Möglichkeiten, eine Applikation von einer Version auf eine spätere zu heben. Dabei geht es um die Möglichkeiten des Upgrades auf direktem Weg (wenn eine oder mehrere Versionen im Upgrade ausgelassen wurden) und die erforderlichen Versionen, die jedenfalls installiert werden müssen. Upgradepfade müssen dokumentiert sein.

Zielgruppen sind

* Entwicklung
* Applikationsmanagement
* Betrieb
* Betrieb des Kunden
* Kunde

Die Dokumentation der Upgradepfade liegt im Allgemeine in einem ausdruckbaren Dokument (z. B. PDF) vor.

#### Besc**h**reibung des Q-Kriteriums

1. Existenz der Beschreibung
2. Inhaltliche Korrektheit
3. Nachweis der Nutzung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität | x |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit | x |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) der Dokumentation (für A.)
* Inhaltliches Review der Dokumentation (für B.)
* Überprüfung der Nachweise der Nutzung (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz Existenz der Dokumentation

Inhaltliche Korrektheit Die Dokumentation ist vollständig und wurde inhaltlich für richtig befunden.

Nachweis der Nutzung Bei der Planung von Aktivitäten im Software Life Cycle werden die Upgradepfade als Entscheidungskriterium für die Optimierung verwendet.

#### Guidance

Die Dokumentation der Upgradepfade stellt einen wesentlichen Teil der Dokumentation der Kompatibilität und maßgebliche Informationen für die Weiterentwicklung und deren Planung dar. Mit der Nutzung der Informationen können mittelfristige Entscheidungen für Kosteneinsparung getroffen, Risiko- und Aufwandsreduktion erzielt werden.

**Beispiel:**

Die Applikation liegt in der Version 2.4 vor. Einige Kunden verwenden noch die Versionen 2.0, 2.1 und 2.2. Die Upgradepfade sind 2.0 -> 2.1 und 2.2 -> 2.3 und 2.2 -> 2.4. Dies bedeutet, dass alle Kunden, die Version 2.1 oder früher verwenden, in jedem Fall zunächst die Version 2.1 installieren müssen, um dann von Version 2.1 auf Version 2.2 upgraden zu können. In der Folge können sie von Version 2.2 auf Version 2.3 oder direkt auf Version 2.4 upgraden, ohne einen Zwischenschritt tätigen zu müssen.

#### Abhängigkeiten

* Release Schein (siehe 3.1.5)
* Anforderungsspezifikation (siehe 3.2.1)

### Richtlinien für die Software-Entwicklung

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Richtlinien helfen der Standardisierung der Systeme und Prozesse und sammeln die relevanten, von der Unternehmensleitung vorgegebenen Regeln an einem zentralen Ort. Die Einhaltung von Richtlinien muss jedenfalls stichprobenartig geprüft werden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Die Artefakte „Richtlinien“ beschreiben die Regeln für einzelne Prozessschritte im Vorgehensmodell. Die Themen sind üblicherweise

* Informationssicherheit
* Architekturvorgaben
* Coding Rules (siehe 3.1.8)
* Layerstruktur der Software
* Qualitätsanforderungen an Testvorgehen
* Datenschutzbestimmungen
* Erfassung gesetzlicher Vorgaben
* Vorgehensmodell für die Software-Entwicklung (Prozess)
* Nutzung von Tools
* Erforderliche Automatisierung

Zielgruppen sind

* Entwicklung
* IT-Architektur
* Testteam

Die Richtlinien liegen üblicherweise in ausdruckbarer Form (z. B. PDF) vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Eignung und Umsetzbarkeit
4. Nachweis der Einhaltung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Überprüfung der Nachweise (Protokolle) (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Richtlinien in unveränderbarer Form

Existenz Mindestinhalt: Mindestinhalt der Richtlinien

Eignung/Umsetzbarkeit: Review mit den Kriterien  
- Eignung für die Richtlinien der ITSV GmbH und ihrer Kunden  
- Überprüfbarkeit  
- Umsetzbarkeit (mit vertretbarem Aufwand)

Einhaltung: Review mit den Inhalten  
- Werden die Vorgaben eingehalten?  
- Sind die Nachweise vorhanden und die Grenzwerte erreicht?

#### Guidance

Die Einhaltung von Richtlinien muss jedenfalls stichprobenartig geprüft werden. Abweichungen von Richtlinien (Tailoring) sind dokumentations- und genehmigungspflichtig.

#### Abhängigkeiten

* Unternehmensziele
* Rahmenverträge mit Kunden

### Coding Rules

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Coding Rules dienen der Vereinheitlichung des Source Codes. Sie sind eine Detaillierung der Richtlinien für die Software-Entwicklung. Je nach definierter Regel kann dies einen positiven Beitrag zur Zuverlässigkeit, Wartbarkeit, Kompatibilität, Sicherheit liefern.

#### Beschreibung des Themas/Artefakts/der Aktivität

Dieses Vorgabedokument beschreibt die Vorgaben an den Code (z. B. Formatierung des Source Codes, erlaubte Sprachkonstrukte) und soll konsistent mit den Prüfungen in der Statischen Analyse der Unit Tests (siehe 3.4.5) sein.

Zielgruppen sind

* Entwicklung
* Tester für Unit Tests
* Reviewer für Code Reviews
* Auditoren (z. B. für Informationssicherheit)

Die Coding Rules müssen in unveränderbarer revisionssicherer Form (z. B. PDF) vorliegen und können gleichlautend in elektronischer Form (z. B. WIKI) verfügbar sein. Eine Implementierung einiger Regeln in „intelligente“ Codeeditoren ist ebenfalls möglich.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Eignung und Umsetzbarkeit
4. Nachweis der Einhaltung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit | x |
| Sicherheit |  |
| Wartbarkeit | x |
| Portierbarkeit | x |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Überprüfung der Nachweise (Protokolle) (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Coding Rules in unveränderbarer Form, kann (teilweise) automatisiert werden

Existenz Mindestinhalt: Sichtkontrolle elementarer Vorgaben (z. B. Standard Coding Rules aus dem Internet)

Eignung/Umsetzbarkeit: Review mit den Kriterien  
- Eignung für die Anforderungen der ITSV GmbH und ihrer Kunden  
- Überprüfbarkeit  
- Umsetzbarkeit (mit vertretbarem Aufwand)

Einhaltung: Review mit den Inhalten  
- Werden die Vorgaben eingehalten?  
- Sind die Nachweise vorhanden und die Grenzwerte erreicht?

#### Guidance

Coding Rules sind essentiell, um Code leicht lesbar zu machen und insbesondere auch innerhalb einer Organisation die Codestruktur zu vereinheitlichen. Übliche Themen für Coding Rules sind

* Formatvorgaben für Code
* Kommentierungsvorgaben für Code
* Länge von Funktionen/Methoden
* Erlaubte Sprachkonstrukte
* Verbotene Sprachkonstrukture
* Vorgaben zur zyklomatischen Komplexität (z. B. McCabe)
* Regeln für die Programmierung von Schnittstellen
* Regeln für defensive Programmierung
* Vorgaben für Validierung von Eingabefeldern

Es ist jedenfalls sinnvoll, die Coding Rules in den betroffenen Teams abzustimmen und vorschlagen zu lassen, um die Akzeptanz zu erhöhen. Freigabe und Entscheidung erfolgt üblicherweise durch die Verantwortlichen für die Informationssicherheit, die Entwicklungsleitung und die Verantwortlichen für das Qualitätsmanagement.

Die Überprüfung von Coding Rules muss mindestens stichprobenartig erfolgen. Üblicherweise werden die Regeln in der statischen Analyse automatisch geprüft. Die manuelle Überprüfung ist nur in Ausnahmen wirtschaftlich vertretbar.

Abweichungen von den Coding Rules müssen dokumentiert begründet und genehmigt werden.

„Altlasten“, d. h. Code, der vor der Einführung der Coding Rules entstanden ist, soll sukzessive adaptiert werden, wobei hier eine konsequente Planung aus wirtschaftlicher Sicht notwendig ist.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Richtlinien für die Software Entwicklung (siehe 3.1.7)

### Fachliches Objektmodell

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die übergreifende Sicht auf das fachliche Objektmodell trägt zur Umsetzung der Enterprise Architecture dar. Ein möglichst redundanzfreies fachliches Objektmodell erhöht die Qualität der Objekte und reduziert Aufwände in der Datenhaltung. Dafür erhöht es die Anforderungen an Zuverlässigkeit und Verfügbarkeit sowie Sicherheit.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Thema fachliches Objektmodell behandelt die Beschreibung der aus Geschäftsprozesssicht erforderlichen Objekte und ihrer Verwendung innerhalb der Gesamtsicht.

Inhalte sind

* Beschreibung der fachlichen Objekte und ihrer Abhängigkeiten
* Referenzierung der verwendenden Stellen
* Bei Duplizierung Darstellung, wo sich der Master befindet

Zielgruppen sind

* Entwicklung
* Betrieb
* IT-Architektur
* Testteam
* Produktmanagement
* Kunde
* Product Owner

Das fachliche Objektmodell liegt üblicherweise in elektronischer Form (z. B. WIKI oder innerhalb eines Tools) vor. Es kann auch in ausdruckbarer Form (z. B. PDF) vorliegen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Berücksichtigung in den Anforderungen und im Design
5. Nachweis der Umsetzung
6. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Überprüfung der Nachweise (Protokolle) (für D.)
* Überprüfung der Nachweise (Protokolle) (für E.)
* Inhaltliches Review (für F.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des fachlichen Objektmodells in dokumentierter Form, kann (teilweise) automatisiert werden

Mindestinhalte: Vorhandensein des fachlichen Objektmodells in aktueller unveränderbarer Form mit den definierten Mindestinhalten, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Korrektheit der beschriebenen Information

Berücksichtigung: Review der Anforderungsspezifikation und des Designdokuments mit den Kriterien  
- Sind alle Aspekte aus dem fachlichen Objektmodell in der Anforderungsspezifikation und im Design berücksichtigt (Traceability)

Einhaltung: Review der Umsetzung (ggf. Test) mit den Kriterien  
- Sind alle Aspekte aus des fachlichen Objektmodells in der Umsetzung berücksichtigt (Traceability)

Vollständigkeit: Review mit den Kriterien  
- Sind alle fachlichen Objekte in der Dokumentation mit den Abhängigkeiten vorhanden. (Traceability)

#### Guidance

Ein zentrales fachliches Objektmodell kann bottom up aufgebaut werden. Zunächst werden die fachlichen Objekte jedes einzelnen Produkts dokumentiert und beschrieben, danach werden die Beschreibungen zusammengeführt und bei Bedarf Redundanzen entfernt (Redesign von Produkten!).

#### Abhängigkeiten

* IT-Map (siehe 3.1.3)
* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe **Fehler! Verweisquelle konnte nicht gefunden werden.**)

### Datenmodell

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die übergreifende Sicht auf das Datenmodell der ITSV GmbH trägt zur Umsetzung der Enterprise Architecture dar. Ein möglichst redundanzfreies Datenmodell erhöht die Datenqualität und reduziert Aufwände in der Datenhaltung. Dafür erhöht es die Anforderungen an Zuverlässigkeit und Verfügbarkeit sowie Sicherheit.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Thema Datenmodell behandelt die Beschreibung der Datenstrukturen und ihrer Verwendung innerhalb der Gesamtsicht der ITSV GmbH. Inhalte sind

* Format und Beschreibung der verwendeten Datenstrukturen
* Referenzierung der verwendenden Stellen
* Bei Duplizierung Darstellung, wo der Master sich befindet

Zielgruppen sind

* Entwicklung
* Betrieb
* IT-Architektur
* Testteam
* Produktmanagement

Das Datenmodell liegt üblicherweise in elektronischer Form (z. B. WIKI oder innerhalb eines Tools) vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Berücksichtigung im Design
5. Nachweis der Umsetzung
6. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Überprüfung der Nachweise (Protokolle) (für D.)
* Überprüfung der Nachweise (Protokolle) (für E.)
* Inhaltliches Review (für F.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Datenmodells in dokumentierter Form, kann (teilweise) automatisiert werden

Mindestinhalte: Vorhandensein des Datenmodells in aktueller unveränderbarer Form mit den definierten Mindestinhalten, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Korrektheit der beschriebenen Information

Berücksichtigung: Review des Designdokuments mit den Kriterien  
- Sind alle Aspekte aus dem Datenmodell im Design berücksichtigt (Traceability)

Einhaltung: Review der Umsetzung (ggf. Test) mit den Kriterien  
- Sind alle Aspekte aus des Datenmodells in der Umsetzung berücksichtigt (Traceability)

Vollständigkeit: Review mit den Kriterien  
- Sind alle Datenstrukturen in der Dokumentation mit den Abhängigkeiten vorhanden. (Traceability)

#### Guidance

Ein zentrales Datenmodell kann bottom up aufgebaut werden. Zunächst werden die Datenstrukturen jedes einzelnen Produkts dokumentiert und beschrieben, danach werden die Beschreibungen zusammengeführt und bei Bedarf Redundanzen entfern (Redesign von Produkten!).

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)
* Fachliches Objektmodell (siehe 3.1.9)

### Standardisierte Lösungen/Komponenten

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit standardisierten Lösungen/Komponenten werden wiederkehrende Aufgaben vereinheitlicht. Diese zentral zur Verfügung gestellten Lösungen reduzieren das Fehlerrisiko und den Entwicklungsaufwand. Andererseits haben sie erhöhte Anforderungen an Verfügbarkeit, Zuverlässigkeit und Flexibilität zu erfüllen.

#### Beschreibung des Themas/Artefakts/der Aktivität

Standardisierte Lösungen/Komponenten können sein:

* Fehlerbehandlung
* Fachliches Logging
* Logging
* Historisierung

#### Beschreibung des Q-Kriteriums

1. Existenz der Entscheidung
2. Berücksichtigung im Design
3. Nachweis der Nutzung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz | x |
| Kompatibilität | x |
| Benutzbarkeit |  |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit | x |
| Portierbarkeit | x |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) der Vorgabe (für A.)
* Inhaltliches Review der Vorhabensdokumente (für B.)
* Test (für C.)
* Code Review (als Teil des Unittests) (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Die Vorgabe muss zentral definiert sein.

Berücksichtigung: Review mit den Kriterien  
- Sind die standardisierten Lösungen/Komponenten berücksichtigt und werden sie verwendet

Nutzung: Grad der Nutzung der standardisierten Lösungen/Komponenten vs. individuelle Lösungen

#### Guidance

An dieser Stelle geht es um die Nutzung von standardisierten Lösungen/Komponenten, nicht die Erstellung. Die Erstellung von standardisierten Lösungen/Komponenten muss beim Design der Lösung entschieden werden.

Standardisierte Lösungen/Komponenten erfordern neben der technischen Umsetzung auf organisatorische Maßnahmen. Im Sinne eines Produktmanagements muss für standardisierte Lösungen/Komponenten die Möglichkeit bestehen, zentral und schnell neue Anforderungen bereitzustellen und dabei die Kompatibilität zu früheren Versionen zu wahren.

Zusätzlich müssen standardisierte Lösungen/Komponenten konsequent dokumentiert, gepflegt und aktualisiert werden, um mit den Erkenntnissen aus anderen Vorhaben/Produkten die Fehlerwahrscheinlichkeit für alle zu reduzieren.

Im Rahmen der Zentralisierung können damit auch Kriterien der Usability behandelt werden (z. B. „lesbare Logfiles“)

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)

## Anforderung definieren

### Anforderungsspezifikation (gemäß EDV Handbuch)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit dem Artefakt Anforderungsspezifikation wird der Scope des Vorhabens festgelegt, und es werden die inhaltlichen Anforderungen (funktional und nicht funktional) festgelegt. Die Anforderungsspezifikation ist Vertragsbestandteil und damit ein bindendes Artefakt. Es wird für die Abnahme seitens des Kunden herangezogen.

#### Beschreibung des Themas/Artefakts/der Aktivität

Die Anforderungsspezifikation gibt es sowohl in „klassischen“ Vorgehensmodellen (z. B. V-Modell) als auch in agilen Vorgehensmodellen. In jedem Fall müssen im Vorfeld einer Entwicklung eines Produkts die Ziele, die Abgrenzung und die groben Inhalte festgelegt werden. In der Anforderungsspezifikation, die auch iterativ entstehen kann, müssen alle Anforderungen festgelegt sein, welche am Ende in der Abnahme überprüft werden. Ob es sich um ein einzelnes Artefakt oder eine Gruppe von Artefakten handelt, wird hier nicht festgelegt.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz | x |
| Kompatibilität | x |
| Benutzbarkeit | x |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit | x |
| Portierbarkeit | x |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer unveränderbaren Form, die von Kunde und ITSV GmbH formal akzeptiert wurde; kann automatisiert werden

Existenz Mindestinhalt: Review der ausgefüllten Struktur gemäß Vorlage (TBD)

Inhaltliche Korrektheit: Review mit den definierten Kriterien (siehe 5.4)

Vollständigkeit: Review mit den definierten Kriterien (siehe 5.5)

*Tailoring*: Tailoring der Mindestinhalte (Reduktion) muss dokumentiert begründet werden und vom Produktverantwortlichen und dem zentralen Qualitäts­management genehmigt werden.

#### Guidance

Das Artefakt Anforderungsspezifikation ist die zentrale Tätigkeit für ein Entwicklungsvorhaben. An dieser Stelle werden die Anforderungen und Erwartungen des Kunden erfasst und für die Umsetzung aufbereitet. Ungenaue und unvollständige Anforderungen führen in vielen Fällen zu Nachfragen im Laufe der Entwicklung oder zu Diskrepanzen zwischen Kunden und ITSV GmbH bei der Abnahme. Daher ist ein gemeinsames Verständnis der Anforderungen für den Erfolg des Vorhabens essentiell. Es gibt Fälle, in denen der Kunde nicht bereit oder nicht in der Lage ist, die Anforderungen zu spezifizieren. In diesen Fällen ist eine gemeinsame Erarbeitung der Anforderungen ggf. auch unter der Leitung der ITSV GmbH erforderlich.

Manchmal sind auch übergreifende Anforderungen, z. B. an den Entwicklungsprozess, an die Struktur von Software, in Richtlinien enthalten.

*Hinweis*: Häufig wird der Ausdruck „Anforderungsspezifikation“ für die Tätigkeit „Anforderungen spezifizieren“ verwendet. Diese Bedeutung ist in diesem Dokument an *keiner* Stelle gemeint.

#### Abhängigkeiten

* Fachliches Objektmodell (siehe 3.1.9)
* EPIC/User Stories (siehe 3.2.2)
* Testrahmenplan (siehe 3.4.1)
* Systemarchitekturkonzept (siehe 3.2.5)
* Richtlinien für die Software-Entwicklung (siehe 3.1.7)

### EPIC/User Stories

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

EPIC/User Stories sind ein Teilthema der Anforderungsspezifikation. Mit ihnen werden die funktionalen Anforderungen an ein Produkt definiert und insbesondere detailliert. Werden EPIC/User Stories formal sinnvoll (z. B. Satzschablonen, vollständig) erstellt, können sie ebenfalls als logische Testfälle verwendet werden. Dann ist lediglich die Definition der Testdaten und des erwarteten Ergebnisses für konkrete Testfälle erforderlich.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Artefakte definieren die funktionalen Anforderungen an eine Applikation. Sie sind eine Detaillierung der Anforderungsspezifikation (siehe 3.2.1) und des Umsetzungskonzepts (siehe 3.2.4).

Zielgruppen sind

* Product Owner
* Entwicklung
* Testteam
* Kunde

Die Artefakte liegen üblicherweise in elektronische Form in einem Requirements Engineering Tool vor. Bei automatischer Generierung von Software können sie auch im entsprechenden Tool zur Verfügung stehen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Inhaltliche Korrektheit
3. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit | x |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer unveränderbaren Form, die von Kunde und ITSV GmbH formal akzeptiert wurde; kann automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Ergeben die EPIC/User Stories einen End-to-End-Prozess?  
- Sind die EPIC/User Stories für die Entwicklung ausreichend detailliert?  
- Decken die EPIC/User Stories Fehlerfälle der Benutzer ab?  
- Sind die EPIC/User Stories widerspruchsfrei?

Vollständigkeit: Review mit den Kriterien  
- Sind alle definierten Geschäftsprozesse abgedeckt?  
- Sind gleichartige EPIC/User Stories identifiziert und zusammengefasst?  
- Sind die EPIC/User Stories vollständig und widerspruchsfrei in Bezug auf die Anforderungsspezifikation?

*Tailoring*: TBD

#### Guidance

Mit der sorgfältigen Definition der EPIC/User Stories zwischen Kunden und ITSV GmbH wird das gemeinsame Verständnis gefördert. Gleichzeitig werden Missverständnis ausgeräumt und die Testfälle für die Abnahme vorbereitet.

Formal ist es sinnvoll, für EPIC/User Stories ein definiertes Schema zu verwenden, das Satzschablonen für die funktionalen Anforderungen verwendet und Abnahmekriterien, erforderliche Daten, erwartete Ergebnisse und insbesondere auch Benutzeroberflächen enthält.

Die Erstellung von EPIC/User Stories enthält wesentliche Teile der Anforderungsanalyse, welche ohnehin erforderlich ist.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Testrahmenplan (siehe 3.4.1)

### Change Request

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit einem Change Request werden Änderungen an Anforderungen in einer geregelten Weise abgewickelt. Wird ein Change Request genehmigt, werden die erforderlichen Änderungen in die Anforderungsspezifikation und das Umsetzungskonzept sowie alle weiteren betroffenen Artefakte aufgenommen. Danach unterliegen die Inhalte des Change Requests den normalen Regeln des Projektmanagements und der Software-Entwicklung.

#### Beschreibung des Themas/Artefakts/der Aktivität

Dieses Artefakt beschreibt den Antrag auf eine Änderung einer bestehenden, vertraglich festlegten Eigenschaft einer Applikation. Inhalte sind

* Eindeutige ID
* Beschreibung der Änderung
* Zu ändernde Anforderungen
* Auswirkungen der Änderung auf das Vorhaben
* Auswirkungen der Änderung auf andere Vorhaben oder System
* Kommerzielle Auswirkungen
* Geplante Zeit
* Betroffene zu ändernde Objekt
* Risiken bei Umsetzung
* Risiken bei Nicht-Umsetzung

Zielgruppen sind

* Interner Auftraggeber (Management)
* Produktmanagement
* Product Owner
* Entwicklung
* Testteam
* Betrieb
* Kunde

Change Requests liegen üblicherweise in ausdruckbarer unveränderbarer Form vor. Alternativ können Change Requests auch in Requirements Engineering Tools verwaltet werden.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Change Request (unabhängig von Medium), kann (teilweise) automatisiert werden

Mindestinhalt: Sind die definierten Mindestinhalte beschrieben?

Inhaltliche Korrektheit: Sind die enthaltenen Informationen korrekt und realistisch?

Vollständigkeit: Sind alle Aspekte der Änderung tatsächlich berücksichtigt und bedacht?

#### Guidance

Inhalte von Änderungen können beispielsweise sein:

* Änderung funktionaler Anforderungen
* Änderung nicht-funktionaler Anforderungen
* Änderung von Konfiguration des Systems
* Änderung eines Service (betrifft üblicherweise das Rechenzentrum)

Formal ist ein Change Request einer Anforderungsspezfikation vor ihrer Freigabe gleichzusetzen, jedoch werden die Planungsaspekte zusätzlich gefordert. Die Erstellung eines guten Change Requests erfordert Überblick über das Vorhaben/das Produkt und Erfahrung im Vorgehen in der Entwicklung und in Vorhaben. Insbesondere die Risikoabschätzung wird häufig nicht mit der notwendigen Sorgfalt durchgeführt, ebenso ist die vollständige Erfassung der betroffenen Lieferobjekte wesentlich, um Inkonsistenzen zu vermeiden.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)

### Umsetzungskonzept

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Das Umsetzungskonzept dient dazu, die fachliche und technische Umsetzung und das Vorgehen bei der Umsetzung zu beschreiben, um so möglichst genau die geplante Lösung darzustellen. Diese Darstellung dient dazu, einerseits dem Entwicklungsteam, andererseits dem Kunden die notwendigen Informationen über die Zusammenhänge bereitzustellen.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Artefakt Umsetzungskonzept stellt die fachlichen und technischen Eigenschaften der Lösung dar. Es ist eine Verfeinerung der Anforderungsspezifikation (siehe 3.2.1), das die Ideen zur Umsetzung darstellt.

Zielgruppen sind

* Entwicklung
* IT-Architektur
* Product Owner
* Testteam
* Betrieb
* Betrieb des Kunden

Üblicherweise liegt das Umsetzungskonzept in unveränderbarer ausdruckbarer Form (z. B. PDF) oder in elektronischer Form in einem Requirements Engineering Tool (z. B. Polarion) vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Berücksichtigung im Design
5. Nachweis der Einhaltung
6. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Inhaltliches/Formales Review (für D.)
* Überprüfung der Nachweise (für E.)
* Inhaltliches/Formales Review (für F.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Umsetzungskonzepts in unveränderbarer Form, kann (teilweise) automatisiert werden

Mindestinhalte: Vorhandensein des Umsetzungskonzepts in aktueller unveränderbarer Form mit den definierten Mindestinhalten, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Korrektheit der beschriebenen Information

Berücksichtigung Design: Review des Designdokuments mit den Kriterien  
- Sind alle Aspekte aus des Umsetzungskonzepts im Design berücksichtigt (Traceability)

Einhaltung: Review der Umsetzung (ggf. Test) mit den Kriterien  
- Sind alle Aspekte aus dem Umsetzungskonzept in der Umsetzung berücksichtigt (Traceability)

Vollständigkeit: Review mit den Kriterien  
- Sind alle Aspekte aus der Anforderungsspezifikation im Umsetzungskonzept berücksichtigt (Traceability)

#### Guidance

Das Umsetzungskonzept entspricht dem Pflichtenheft aus dem V-Modell. Dieses Artefakt kann aus mehreren Dokumenten bestehen, z. B. kann das SAK einen Teil der Beschreibung übernehmen. Für das Umsetzungskonzept ist es essentiell, dass hier das WIE der Umsetzung beschreiben wird (während in der Anforderungsspezifikation das WAS beschrieben wird).

Das Umsetzungskonzept sollte von allen relevanten Stakeholdern akzeptiert sein, um das Einverständnis über die Anforderungen und deren Umsetzung frühzeitig zu erzielen.

In der ITSV GmbH wird die erste Stufe des Umsetzungskonzepts zur Erarbeitung der Inhalte und der Schätzung für ein Angebot an den Kunden erarbeitet. Es ist

Das Umsetzungskonzept kann bei agilen Vorgehensweisen während der Iteration entstehen. In jedem Fall solle eine Baseline zu jeder Produktivsetzung der Entwicklungsergebnisse erreicht werden.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)

### Systemarchitekturkonzept (SAK)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Das Systemarchitekturkonzept beschreibt den technischen Teil des Umsetzungskonzepts. Damit wird festgelegt, wie die technische Umsetzung für ein Vorhaben aussehen soll. Weiters werden im Systemarchitekturkonzept die wesentlichen technischen Randbedingungen in Bezug auf Sicherheit, Architektur, Einbettung in bestehende Systeme u. Ä. dokumentiert, sodass an dieser Stelle die übergeordnete technische Information zentral vorhanden ist.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Systemarchitekturkonzept ist in der ITSV eines der zentralen Dokumente für die Beschreibung des technischen Teils der Lösung und deren Umsetzung. Das Systemarchitekturkonzept wird zu Beginn einer Umsetzung aus der Anforderungsspezifikation (siehe 3.2.1) erstellt und ist in einer ersten Version Grundlage für die Beauftragung der Durchführung. In der Folge wird das Systemarchitekturkonzept zumindest bei der Abnahme aktualisiert und ergänzt.

Im laufenden Betrieb muss das Systemarchitekturkonzept in regelmäßigen Abständen (z. B. alle 2 Jahre) auf Gültigkeit überprüft werden.

Zielgruppen sind

* Produktmanagement
* IT-Architektur
* Lösungsdesign
* Entwicklung
* Applikationsmanagement
* Betrieb
* Interner Auftraggeber (Management)
* Stabsstelle Informationssicherheit
* Stabsstelle Qualität

Das Systemarchitekturkonzept liegt üblicherweise in ausdruckbarer Form (z. B. PDF) vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Formales/inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Systemarchitekturkonzepts in unveränderbarer Form, kann automatisiert werden

Mindestinhalt: Die Mindestinhalte gemäß Vorgabe (siehe TBD) sind enthalten, kann teilweise automatisiert werden.

Inhaltliche Korrektheit: Review mit den Kriterien  
- Sind die enthaltenen Informationen korrekt und konsistent?

Vollständigkeit: Review mit den Kriterien  
- Sind alle geforderten Informationen in der notwendigen Beschreibungstiefe vorhanden?

#### Guidance

Das Systemarchitekturkonzept ist ein grundlegendes Dokument für die Umsetzung von Anforderungen. Als zentrales Dokument enthält es alle wesentlichen Informationen über die Umgebungen, Einstufungen bzgl. Informationssicherheit, Abhängigkeiten zu anderen Applikationen usw. Ein eingehendes Review, um die Korrektheit zu gewährleisten, ist daher notwendig und sinnvoll. Das Systemarchitekturkonzept kann somit als eines der wesentlichen Dokumente im Software Life Cycle als Nachschlagewerk dienen.

Das Systemarchitekturkonzept entsteht – abgesehen von den erforderlichen Mindestinhalten – insbesondere in agilen Vorgehensweisen iterativ und wir sukzessive verfeinert.

Es ist notwendig, dass das Systemarchitekturkonzept von den definierten Kontrollinstanzen geprüft und freigegeben wird, da dieses Dokument eines der zentralen technischen Dokumente für die Umsetzung ist.

Das Systemarchitekturkonzept sollte bei jeder Änderung auf Aktualität und Korrektheit geprüft werden, mindestens jedoch alle 3 Jahre.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* EPIC/User Stories (siehe 3.2.2)
* Umsetzungskonzept (siehe 3.2.4)
* Release Schein (siehe 3.1.5)
* Testrahmenplan (siehe 3.4.1)

### Design for Testability

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Um das korrekte Verhalten einer Applikation prüfen zu können, ist es in vielen Fällen erforderlich, Zwischenergebnisse oder das Programmablaufverhalten beobachten zu können. Hierzu können einerseits im Software-Design, andererseits in der Software-Implementierung unterstützende Maßnahmen gesetzt werden. Die rechtzeitige Implementierung derartiger Maßnahmen reduziert den Testaufwand und erleichtert die Fehlersuche.

Zusätzlich dient das Design for Testability dazu, dass Testfälle mit vertretbarem Aufwand automatisiert werden können.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Thema Design for Testability behandelt die Anforderungen, die an eine Applikation aus Sicht der Testbarkeit gestellt werden. Dieses Detailthema der Anforderungsspezifikation (siehe 3.2.1) definiert die Anforderungen, die von

* Testteam
* Applikationsmanagement

an die Umsetzung gestellt werden, in der Folge erfüllen zu können. Die Anforderungen können aus allen Teststufen (Unit Test, Integrationstest, Systemtest, Akzeptanztest sowie speziellen Testarten) entstehen.

#### Beschreibung des Q-Kriteriums

1. Existenz der Entscheidung
2. Risikobasierte Berücksichtigung
3. Vollständige Berücksichtigung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz | x |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) der grundlegenden Entscheidung (für A.)
* Inhaltliches Review der Designdokumentation (für B.)
* Code Review (als Teil des Unittests) (für B.), kann weitgehend automatisiert werden
* Lessons Learned aus dem Test (für B.)
* Inhaltliches Review der Designdokumentation (für C.)
* Code Review (als Teil des Unittests) (für C.), kann weitgehend automatisiert werden
* Lessons Learned aus dem Test (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz Existenz der Vorgabe für Design for Testability

Risikobasierte Berücks. Es gibt für einzelne hochkritische Applikation oder Teile von Applikationen die Entscheidung zu Design for Testability sowie die Überprüfung der Umsetzung (auf Basis der Anforderungen des Testteams)

Vollständige Berücks.: Es gibt für alle Applikationen die Entscheidung zu Design for Testability sowie die Überprüfung der Umsetzung (auf Basis der Anforderungen des Testteams)

#### Guidance

Eine Entscheidung für Design for Testability sollte bewusst getroffen werden und die Ziele und Erwartungen darlegen. Eine derartige Entscheidung kann in der Umsetzung anfänglich höhere Aufwände erfordern, deren Amortisation in der Folge bewertet werden muss.

Zum Treffen einer risikobasierten Entscheidung ist die Risikoeinstufung der Applikation oder Teile der Applikation erforderlich. Die Entscheidung kann im Software Life Cycle in der Designdokumentation dokumentiert und in ihrer Ausprägung präzisiert werden. Auf Basis der definierten Ziele und Erwartungen müssen dann die zugehörigen Überprüfungen durchgeführt werden.

Es ist sinnvoll, dass das Testteam die Anforderungen bereits in der Anforderungsspezifikation festlegt und detailliert. Diese Anforderungen werden wie alle anderen Anforderungen behandelt und müssen eingehalten werden. Beispiele sind

* Anforderungen an das GUI für die Testautomatisierung (eindeutige ID eines Feldes)
* Abfragefunktionen, um bei Fehlverhalten die Ursache eingrenzen zu können

Die Einhaltung der Umsetzung kann im Code Review (als Teil des Unittests) geprüft werden. Für eine automatische Prüfung bedarf es Vorgaben in den Coding Rules. Zur Umsetzung kann einerseits im Design das Erfordernis von prüfbaren Zwischenergebnissen, andererseits die Instrumentierung des Source Codes mit Zwischenausgaben verwendet werden.

**Beispiel für Berücksichtigung im Design**

Eine Gruppe von zwei Applikationen hat als Ergebnis eine druckbare PDF-Datei. In einer der Applikationen wird die PDF-Sequenz erzeugt, in der anderen Applikation werden formale Adaptionen am Layout in der PDF-Sequenz durchgeführt.

Die die PDF-Sequenz erzeugende Applikation kann als Black Box nur dann mit vertretbarem Aufwand geprüft werden, wenn die Schnittstelle mit abgespeicherten Daten und nicht als Datenstrom realisiert wird.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)
* Testrahmenplan (siehe 3.4.1)

## Anforderung umsetzen

### Software Design

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Das Software Design dient der Verfeinerung der Lösung aus technischer Sicht.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Software Design beschreibt die software-technische Lösung. Es hat typischerweise folgende Inhalte

* Komponentendiagramm
* Technisches Datenmodell (siehe 3.1.10)
* Aktivitätendiagramme
* Klassendiagramme

Zielgruppen sind

* Entwicklung
* Application Management

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit | x |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Formales Review (für D.)
* Inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Software Designs in unveränderbarer Form, kann (teilweise) automatisiert werden

Mindestinhalte: Vorhandensein des Software Designs in aktueller unveränderbarer Form mit den definierten Mindestinhalten, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Korrektheit der beschriebenen Information

Vollständigkeit: Review mit den Kriterien  
- Sind alle es Systems aktuell und vollständig beschrieben?

#### Guidance

Das Software Design soll als aktuelle Referenz für die Entwicklung verwendet werden. Es folgt den üblichen Regeln der Software-Entwicklung (z. B. UML).

Im Software Design, das sinnvollerweise auch formal und inhaltlich überprüft wird, werden die Themen aus der Anforderungsspezifikation und dem Umsetzungskonzeptanalysiert und die Art der Umsetzung beschrieben (z. B. Design Review). Prüfpunkte siehe 5.7)

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)
* Softwarearchitekturkonzept (siehe 3.2.5)
* Datenmodell (siehe 3.1.10)

### Schnittstellendokumentation

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Schnittstellendokumentation dient der Nachvollziehbarkeit und als Referenz für andere Produkte, die auf diese Schnittstellen zugreifen wollen.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Artefakt Schnittstellendokumentation beschreibt die Formate und das Verhalten von Schnittstellen zwischen Komponenten oder Systemen (interne und externe Schnittstellen). Zum Inhalt gehören

* Datenformat der Übertragung
* Verhalten der Schnittstelle im Fehlerfall (inkorrekte oder unvollständige Daten)
* Zeitliches Verhalten der Schnittstelle

Zielgruppen sind

* Entwicklung
* Tester für White Box Tests

Die Schnittstellendokumentation kann in folgenden Formaten vorliegen

* Ausdruckfähiges Dokument (z. B. PDF; sinnvoll für externe Schnittstellen zur Weitergabe)
* Elektronische Dokumentation (z. B. WIKI)
* Source Code Kommentare (nicht empfohlen, außer die Dokumentation ist automatisch extrahierbar)

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Formales Review (für D.)
* Inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Schnittstellendokumentation in unveränderbarer Form, kann (teilweise) automatisiert werden

Mindestinhalte: Vorhandensein der Schnittstellendokumentation in aktueller unveränderbarer Form mit den definierten Mindestinhalten, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Korrektheit der beschriebenen Information

Vollständigkeit: Review mit den Kriterien  
- Sind alle Aspekte der Schnittstelle aktuell und vollständig beschrieben?  
- Zielgruppenorientierung der Beschreibung

#### Guidance

Schnittstellendokumentation ist essentiell für die spätere Wartung eines Produkts. Bereits bei der Implementierung des Produkts durch unterschiedliche Personen/Gruppen ist die aktuelle korrekte Dokumentation erforderlich. Sie wird zusätzlich vom Test genutzt, um Stubs und Treiber für Schnittstellen herzustellen.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)
* Software Design (siehe TBD)
* Anforderungen der Basisprotokolle

### Standard Request Header

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der Standard Request Header dient der eindeutigen Identifizierung einer Applikation (in TR2 und TR3). Ist er vorhanden, kann beim Aufruf einer Funktion (besser) erkannt werden, ob die aufrufende Funktion berechtigt ist, den Request abzusetzen. Der Standard Request Header wurde aus Sicherheitsgründen eingeführt.

#### Beschreibung des Themas/Artefakts/der Aktivität

Der Standard Request Header ist eine definierte Struktur, die beim Aufruf eines Webservice vorhanden sein muss. Diese software-technische Einrichtung wird programmiert, um an der Schnittstelle zum Webservice erforderliche Informationen in definierter Form vorliegen zu haben (z. B. Benutzerkennung, ID der aufrufenden Applikation). Ist der Standard Request Header inkorrekt oder nicht vorhanden, bzw. ist die aufrufende Applikation nicht registriert (zugelassen), wird der Aufruf abgebrochen.

Die Regeln für die Verwendung von Standard Request Headers müssen in der Software Dokumentation zentral festgelegt sein (Schnittstellendefinition). Die Informationen über die den Standard Request Header verwendenden Objekte müssen zentral und aktuell dokumentiert sein.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Inhaltliche Korrektheit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit | x |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Code Review (als Teil des Unittests) (für A.)
* Integrationstest (für A.)
* Code Review (als Teil des Unittests) (für B.)
* Integrationstest (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Die Software muss an den Schnittstellen einen Standard Request Header verwenden.

Inhaltliche Korrektheit: Wird die Funktion ohne oder mit inkorrektem Standard Request Header aufgerufen, muss die Rücklieferung der angeforderten Information verweigert werden.

*Tailoring*: Die Verwendung von Standard Request Headern ist verpflichtend für die Neudefinition von Schnittstellen.   
Wird eine Schnittstelle verändert und besitzt noch keinen Standard Request Header, muss das Weiterbestehen der Ausnahme von der Verpflichtung der Verwendung dokumentiert und vom Prozessverantwortlichen und vom Qualitätsmanagement genehmigt werden.

#### Guidance

Die Prüfung auf Existenz des Standard Request Header ist nur sinnvoll, wenn die inhaltliche Korrektheit für den Standard Request Header gegeben ist. Lediglich bei Vorbereitungsarbeiten zur Implementierung des Standard Request Headers an einer Schnittstelle kann die Existenz vorübergehend ein sinnvolles Q-Kriterium sein.

#### Abhängigkeiten

Coding Rules

SAK

Integrationstest

Code Review (als Teil des Unittests)

Sicherheitanforderungen

### Benutzerdokumentation

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Benutzerdokumentation unterstützt essentiell die Verwendbarkeit von Applikation durch die Zielgruppen. Einfache Verfügbarkeit und leichte Auffindbarkeit von Informationen beschleunigen den Arbeits­vorgang des Endbenutzers in Zweifelsfällen und entlasten Helpdesk und Key-User, indem Nachfragen reduziert werden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Benutzerdokumentation ist die Dokumentation, die den unterschiedlichen Benutzergruppen für das Erlernen der Bedienung und als Referenz bei der Bedienung der Software bereitgestellt wird. Es wird unterschieden zwischen Anleitungsdokumenten (zum Erlernen oder Nachschlagen einzelner häufiger Abläufe) und Referenzdokumenten (zum Nachschlagen einzelner Funktionen).

Zielgruppen sind

* Endbenutzer (SachbearbeiterInnen),
* Administrationspersonal/Personal des Service Desks
* Experten, welche die Möglichkeiten des Systems (für Erweiterungen) nutzen wollen

Benutzerdokumentation soll revisionssicher sein und kann in folgenden Formen vorliegen: Ausdruckbare Form (z. B. PDF), Online-Hilfe, elektronische Form (z. B. WIKI, insbesondere für Administrationspersonal und Experten), visuelle Form (z. B. Video-Tutorials).

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Inhaltliche Korrektheit
3. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)
* Formales/inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Benutzerdokumentation in unveränderbarer Form, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Zielgruppenorientierung der Beschreibung  
- Korrektheit der beschriebenen Abläufe

Vollständigkeit: Review mit den Kriterien  
- Sind alle Abläufe beschrieben?  
- Ausrichtung der Beschreibungsstruktur auf die vorgesehene Nutzungsart  
- Zielgruppenorientierung der Beschreibung

#### Guidance

Der Umfang der Benutzerdokumentation wird im Umsetzungskonzept festgelegt. Soll keine Benutzerdokumentation erstellt werden, wäre es sinnvoll, dies zu begründen.

Möglichkeiten der Benutzerdokumentation sind

* Handbuch aller Benutzerinteraktionen (Referenzdokument)
* Handbuch der vorgesehenen Anwendungsfälle (Anleitungsdokument)
* Handbuch des betrieblichen Objektmodells (für Betrieb und Service Desk)

Für Benutzerdokumentation muss zunächst die Zielgruppe identifiziert werden. Dazu ist es erforderlich die Kenntnisse der Zielgruppe zu kennen und die Beschreibungstiefe darauf anzupassen. Im zweiten Schritt muss entschieden werden, wie die Benutzerdokumentation genutzt werden soll – als Referenzwerk zum Nachschauen von Funktionalitäten und Parametern oder als Einführungswerk in die Applikation. Im ersten Fall ist die Benutzerdokumentation sehr strukturiert aufgebaut, um gezielt Informationen zu finden. Im letzteren Fall ist die Benutzerdokumentation gemäß den typischen Abläufen aufgebaut.

Benutzerdokumentation kann vom Test für die Erstellung von Testfällen verwendet werden (inkl. Automatisierung)

#### Abhängigkeiten

* Umsetzungskonzept (siehe 3.2.4)
* Betriebshandbuch (siehe )

### Release Notes

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Release Notes sind ein zentraler Bestandteil des RDM-Prozesses in der ITSV GmbH. Sie dienen bei der Übergabe aus der Entwicklung an den Betrieb und in der Folge bei der Weiterentwicklung als zentrale Informationsbasis für das Applikationsmanagement, System Management und als Information zu den Kunden. Mit dem Q-Kriterium wird die Qualität der Informationsverteilung überwacht.

#### Beschreibung des Themas/Artefakts/der Aktivität

Release Notes sind ein Dokument, das die Neuerungen, Bestandteile und Fehlerbehebungen einer Software-Release in Bezug auf die Vorgänger-Release auf Basis der priorisierten Anforderungen beschreibt. Dabei gibt es einen fachlichen Teil, der Bezug auf die funktionalen Änderungen und Neuerungen nimmt, und einen technischen/organisatorischen Teil, der darstellt, welche Bestandteile eine Release hat, welche Fehler behoben wurden, Auskunft über Kompatibilität gibt u. Ä.

Zielgruppen sind

* Kunde,
* Zentrale Release-Koordination
* Betrieb
* Personen, die Informationen zum aktuellen Stand einer Software benötigen.

Release Notes liegen im Allgemeinen in ausdruckbarer Form vor (z. B. PDF) um sicherzustellen, dass die Weitergabe an die Zielgruppen unabhängig von speziellen Werkzeugen möglich ist.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer unveränderbaren Form, die von der Zielgruppe gelesen werden kann; kann automatisiert werden

Existenz Mindestinhalt: Review der ausgefüllten Struktur gemäß Vorlage (5.2)

Inhaltliche Korrektheit: Review mit den Kriterien (siehe 5.3)  
- Vollständigkeit gemäß den Vorgaben  
- Korrektheit auf Basis der verfügbaren Entwicklungsinformationen  
- Konsistenz mit Anforderungen an die Umsetzung  
- Konsistenz mit Fehlerdokumentation  
- Konsistenz mit Releaseplanung

*Tailoring*: Der technische Teil der Release Notes ist verpflichtend und unterliegt keiner Möglichkeit zum Tailoring.  
Der fachliche Teil der Release Notes ist verpflichtend, jedoch nicht für alle Produkte erforderlich. Tailoring für den Entfall der verpflichtenden Existenz ist zu dokumentieren und vom Prozessverantwortlichen und von Qualitätsmanagement zu genehmigen

#### Guidance

Die Existenzprüfung der Release Notes ist nur dann ausreichend, wenn trotzdem stichproben­weise die Inhalte geprüft werden. In der ITSV GmbH scheinen einzelne Release Notes über mehrere Jahre konstant zu bleiben, was die Idee der Release Notes ad absurdum führt.

Die Prüfung der Existenz des Mindestinhalts kann entweder im Rahmen der Prüfung auf inhaltliche Korrektheit erfolgen oder getrennt, indem die Struktur des Dokuments (Inhaltsverzeichnis) mit der vorgegebenen Struktur verglichen wird und in den Teilen, die ausgefüllt werden müssen, auch tat­sächlich plausible Information enthalten ist.

Die Prüfung auf inhaltliche Korrektheit ist der eigentlich wesentliche Punkt, da hier ein Übergang in einen anderen Verantwortungsbereich erfolgt. Das Review kann vom Applikationsmanagement, dem Produktmanagement und dem Service Management gemeinsam durchgeführt werden. An dieser Stelle wird auch für den Kunden dokumentiert, welche der ursprünglich vereinbarten und geplanten Anforderungen und Fehler umgesetzt bzw. behoben wurden, um mit dieser Information spätere Nachfragen zu vermeiden.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Systemarchitekturkonzept (siehe 3.2.5)
* Testprozess (siehe 3.4)

## Testprozess

Details zum Testprozess und seiner Teststufen sind in den Unterlagen der ITSV GmbH beschrieben (siehe TBD).

### Testrahmenplan

#### Beschreibung des Themas/Artefakts/der Aktivität

Der Testrahmenplan definiert, welche Testaktivitäten für welche Art von Vorhaben durchgeführt werden. Er enthält die Beschreibung der konkreten Testumgebungen sowie die Rollen und Verantwortungen. Zielgruppen sind

* Produktmanagement
* Product Owner
* Entwicklung
* Testteam
* Betrieb
* Kunde

Der Testrahmenplan liegt üblicherweise in ausdruckbarer Form (z. B. PDF) vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Übereinstimmung mit Anforderungen und Erwartungen
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der Testrahmenplan dient der Nachvollziehbarkeit der geplanten Testschritte und der (detaillierten) methodischen Planung des Tests.

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)
* Formales/inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Testrahmenplans in unveränderbarer Form, kann (teilweise) automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Angemessenheit der Tests für das Schutzbedürfnis  
- Korrektheit der beschriebenen Abläufe  
- Angemessenheit der Tests für die Anforderungen

Vollständigkeit: Review mit den Kriterien  
- Sind alle Tests detailliert beschrieben?

#### Guidance

Der Testrahmenplan ist das zentrale Testplanungsdokument. Es legt die Vorgehensweise auf Basis der Anforderungen fest. Üblicherweise definiert dieses Dokument auch die Testendekriterien (wann der Test erfolgreich war), die Testumgebungen und vor allem die Testziele.

Der Testrahmenplan soll alle Teststufen und die Methoden ihrer Umsetzung beschreiben. Dies umfasst auch Vorgaben über die funktionale Abdeckung der Anforderungen durch Tests (z. B. mindestens ein Positiv- und ein Negativ-Testfall pro Anforderungen, „das Idealszenario der user Story muss mit einem Testfallabgedeckt sein“).

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Systemarchitekturkonzept (siehe 3.2.5)
* Software Design (siehe 3.3.1)
* Design for Testability (siehe 3.2.6)

### Testplan

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit dem Testplan wird der Bereich „Test“ einer Entwicklung gemäß Projektmanagementvorgehen geplant und in der Folge gesteuert. Mit einem Testplan ist einerseits die inhaltliche Verbindung zum Testrahmenplan, andererseits die planerische Verbindung zu den Aktivitäten der Umsetzung gewährleistet und nachvollziehbar.

#### Beschreibung des Themas/Artefakts/der Aktivität

Dieses Artefakt ist der Zeitplan, in welchem die im Testrahmenplan definierten Tests ablaufen. Er ist ein Artefakt des klassischen Projektmanagements und enthält neben dem zeitlichen Ablauf die Aktivitäten und Verantwortlichen sowie die erwarteten Ergebnisse. Zielgruppen sind

* Produktmanagement
* Product Owner
* Entwicklung
* Testteam
* Kunde

Der Testplan liegt üblicherweise in elektronischer Form (z. B. MS Project, EXCEL) oder in einem Tool (z. B. Polarion) vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer revisionssicheren Form, kann automatisiert werden.

Existenz Mindestinhalt: Review der ausgefüllten Struktur gemäß den im Testrahmenplan definierten Aktivitäten, kann automatisiert werden.

Inhaltliche Korrektheit: Review mit den Kriterien   
- Vollständigkeit der Aktivitäten  
- Plausibilität der Durchlaufzeiten  
- Konsistenz mit Gesamtplanung  
- Konsistenz mit Releaseplanung  
- Konsistenz mit verlautbarten Terminen  
Kann bei geeigneter Dokumentenplanung und Toolnutzung automatisiert werden.

*Tailoring*: Der Testplan ist verpflichtend und unterliegt keiner Möglichkeit zum Tailoring.

#### Guidance

Der Testplan ist die Detaillierung des Testrahmenplans und definiert den Projektplan für das Teilprojekt „Test“. Üblicherweise werden – abhängig von der Verantwortung für die einzelnen Teststufen – Integrationstests, Systemtests und Abnahmetests sowie Last- und Performancetests und Securitytests im Testplan erfasst. Die erfolgt unter der Annahme, dass Unit Test in der Verantwortung der Entwicklung liegen und nicht von einer unabhängigen Gruppe durchgeführt werden.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)

### Testnachweise (Testprotokolle)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit dem Nachweis von Tests kann die Einhaltung der Sorgfaltspflicht belegt werden. Tests werden auf unterschiedlichen Ebenen gefordert: ISO 9001, ISO 27001, Produkthaftungsgesetz, Vorgaben des Hauptverbandes der österreichischen Sozialversicherung. Die produktive Verwendung von Software ohne *nachweisliche* Tests kann daher im Schadensfall sogar einen Gesetzesverstoß darstellen (Compliance).

Der Grad der Abdeckung der Tests ist ein maßgeblicher Input für die Risikoabschätzung im laufenden Betrieb.

#### Beschreibung des Themas/Artefakts/der Aktivität

Testnachweise sind die Aufzeichnungen und Dokumentation der Test im Lebenszyklus. Testnachweise bestehen aus

* Testrahmenplan (siehe 3.4.1)
* Testplan für jede Teststufe (siehe 3.4.2)
* Definition der der Testfälle
* Definition der Testdaten
* Protokollierung der Testdurchführung mit Ergebnis, Datum und durchführender Person

Zielgruppen sind

* Entwicklung
* Applikationsmanagement
* Product Owner
* Kunde
* Auditoren

Die Testnachweise liegen im Allgemeinen in ausdruckbarer Form (z. B. PDF) bei Planungsdokumenten und elektronischer Form (z. B. in einem Tool) für die Testfälle, Testdaten und Testergebnisse vor.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Testnachweise (kann automatisiert werden)

Grad der Testabdeckung: Review der Testnachweise mit den Kriterien  
- Wie viele der Anforderungen wurden durch Tests auf Korrektheit geprüft?

*Tailoring*: Testnachweise sind verpflichtend. Ob Tests überhaupt und in welcher Tiefe sie dokumentiert werden müssen, wird im Testrahmenplan festgelegt.

#### Guidance

Um Tests nachweisen zu können, müssen die Anforderungen und Erwartungen an das Testobjekt geklärt sein. Üblicher Weise sind Tests für die Funktionalität unbedingt erforderlich, ebenso Tests für Informationssicherheit, sofern personenbezogene Daten verarbeitet werden.

Der Nachweis der Tests kann in unterschiedlicher Form erfolgen: In einem Tool, in dem die Tests durchgeführt werden (Grad der Abdeckung ist hier auch zumindest intuitiv nachweisbar), auf Papier in einem Testprotokoll, das unterschrieben und datiert ist und die Testergebnisse festhält, auf Papier für jeden einzelnen Test (unterschrieben und datiert mit Ergebnis).

Die Unterschrift bestätigt die Testdurchführung und dokumentiert damit die übernommene Verantwortung der testenden Person.

Tests, die nicht erfolgreich waren, müssen bewertet werden (Risiko).

Die Testnachweise umfassen den Stand (Konfiguration) der Testumgebung, auf welcher getestet wird, inkl. Versionsnummer der Software, Build-Nummer, verwendete Server usw.

Testnachweise sind die wesentlichen Grundlagen für Quality Gates.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)
* Testplan (siehe 3.4.2)

### Testautomatisierung

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der Nutzen der Testautomatisierung liegt in der einfachen Wiederholbarkeit von definierten Test ohne manuelle Interaktion. Mit der Testautomatisierung werden Aufwände insbesondere für Regressionstests reduziert.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität automatisiert ausgewählte Testfälle und bereitet Testdaten für die automatische Ausführung dieser Testfälle auf. Sie basiert auf den klassischen Schritten des Testmanagements. Artefakte sind

* Testrahmenplan (siehe 3.4.1)
* Testplan (siehe 3.4.2)
* Testfälle
* Testdaten
* Testnachweise (siehe 3.4.3)

Zielgruppen sind

* Produktmanagement
* Product Owner
* Testteam
* Betrieb

Zum Vorliegen der einzelnen Artefakte siehe die jeweiligen Beschreibungen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Vergleich der geplanten automatisierten Testfälle mit den tatsächlich aktuell und funktionieren umgesetzten Testfällen (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Planung der Testautomatisierung für ein Produkt gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Automatisier.: Review der automatisierten Testfälle in Bezug auf den vorgegebenen Grad der Abdeckung durch Testautomatisierung.

*Tailoring*: Eine Entscheidung über den Grad der Testautomatisierung muss in jedem Fall getroffen werden.

#### Guidance

Testautomatisierung ist sinnvoll, wenn Testfälle innerhalb eines einigermaßen stabilen Systems wiederholt (mehr als 5 bis 7 Mal) ausgeführt werden müssen. Bei der Erstellung der Testfälle muss auf die Automatisierbarkeit geachtet werden, ebenso müssen die Testdaten definiert werden. Für die Ausführung ist ein definierter Zustand der Testumgebung erforderlich (Testvoraussetzungen).

Testautomatisierung erfordert nach der Erstellung auch Wartungsaufwand, um die Testfälle an Änderungen und ggf. auch an Änderungen der Infrastruktur anzupassen.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)
* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)

### Unit Test

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die Aktivität Unit Test ist für die Implementierung von Software die erste Teststufe. Sie soll sicherstellen, dass die einzelnen Units einerseits funktional korrekt arbeiten (Black Box Sicht), andererseits strukturell die definierten Anforderungen erfüllen (z. B. Coding Rules) (White Box Sicht). Erfolgreiche umfassende Unit Test können die Integration erleichtern und die spätere Fehlersuche im integrierten System reduzieren.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität überprüft Funktionalität und nichtfunktionale Eigenschaften einzelner Units (z. B. Software-Module). Unit Tests können dynamisch und statisch sein und dienen im Wesentlichen der Verifikation der Anforderungen.

Unit Test umfassen folgende Methoden:

* Statische Analyse
* Code Review
* Tests (dynamische White Box Tests)

Zur Nachweisbarkeit der Tests siehe 3.4.3.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit | x |
| Wartbarkeit | x |
| Portierbarkeit | x |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Unit Tests gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der Unit Tests in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Unit Tests sind verpflichtend (außer für SAP). Tailoring für den Entfall der Dokumentation der Testergebnisse ist zu dokumentieren und vom Prozessverantwortlichen und von Qualitätsmanagement zu genehmigen. Tailoring für den Entfall von Unit Tests ist zu dokumentieren, zu begründen und vom Prozessverantwortlichen und von Qualitätsmanagement zu genehmigen.

#### Guidance

Unit Tests können viele Ausprägungen haben. Der erste Aspekt sind die statischen Tests, welche sich mit der Struktur des Source Codes auseinandersetzen. Hierbei werden die Vorgaben an die Codestruktur überprüft. Dies kann in Code Reviews (manuell) für spezielle Kriterien (z. B. hohe Kritikalität des Codeteils, bei automatisierten dynamischen Tests nicht erreichbare Codeteile) oder automatisiert durchgeführt werden (Nutzung von Tools zur statischen Analyse, die auf die Coding Rules ausgerichtet sind und die dort definierten Kriterien überprüfen. Der zweite Aspekt ist die dynamische Ausführung, in der einerseits die Funktionalität überprüft wird, andererseits aber auch die vorgegebene Codeabdeckung erreicht werden soll. Man unterscheidet hier zwischen Statement-, Decision- und Path-Coverage. Relevant ist üblicherweise die Statement-Coverage, manchmal auch die Decision-Coverage. Path-Coverage ist in nahezu allen Fällen aufgrund der erforderlichen Aufwände wirtschaftlich nicht vertretbar.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)
* Software Design (siehe 3.3.1)

### Integrationstest

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die Aktivität Integrationstest ist für die Implementierung von Software die zweite Teststufe. Sie soll sicherstellen, dass die einzelnen Units, die nach erfolgreichen Unit Tests als fehlerfrei angenommen werden, korrekt zusammenarbeiten und korrekt über die Schnittstellen kommunizieren. Erfolgreiche umfassende Integrationstests können die Fertigstellung des Produkts erleichtern und die spätere Fehlersuche im Produkt reduzieren.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität dient der Zusammenführung von mehreren Komponenten (Modulen). Ziel ist es, das Zusammenwirken der Komponenten zu prüfen, nicht die eigentliche Funktionalität der einzelnen Komponenten.

Integrationstests sind üblicherweise Schnittstellentests.

Zur Nachweisbarkeit der Tests siehe 3.4.3.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Integrationstests gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der Integrationstests in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Integrationstests sind verpflichtend (außer für SAP). Tailoring für den Entfall der Dokumentation der Testergebnisse ist zu dokumentieren und vom Prozessverantwortlichen und von Qualitätsmanagement zu genehmigen. Tailoring für den Entfall von Integrationstests ist zu dokumentieren, zu begründen und vom Prozessverantwortlichen und von Qualitätsmanagement zu genehmigen.

#### Guidance

Integrationstests sind dynamische Tests, bei denen geprüft wird, ob einzelne Units korrekt miteinander kommunizieren. Dabei sollte vorher geplant werden, wie viele Units auf einmal integriert werden sollen, ob die Integration bottom up oder top down erfolgt. In ausgewählten Fällen kann auch ein „Big Bang“ sinnvoll sein. Außer im Falle des „Big Bang“ sind zusätzliche Maßnahmen (i.A. Software) erforderlich, welche die Schnittstellen zu nicht für diese Integration vorgesehenen Units simulieren (Input und Output). Diese Software sollte von anderen Personen erstellt werden als die Units selbst, um systematische Fehler zu vermeiden.

Für die Testnachweise muss die Konfiguration des Testsystems beschreiben sein.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Software Design (siehe 3.3.1)
* Schnittstellendokumentation (siehe 3.3.2)
* Testrahmenplan (siehe 3.4.1)

### Systemtest

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die Aktivität Systemtest ist für die Fertigstellung von Software die letzte interne Teststufe der ITSV GmbH. Sie soll darstellen, dass das Gesamtsystem gemäß den Erwartungen und Vorgaben funktioniert und nachweisen, dass alle vereinbarten Anforderungen umgesetzt wurden. Erfolgreiche umfassende Systemtests können die Abnahme des Produkts erleichtern und Fehler in der Produktion reduzieren.

Der Systemtest soll die Erfüllung der High Level Abnahmekriterien des Umsetzungskonzepts (siehe 3.2.4), wie sie mit dem Kunden vereinbart sind, nachweisen.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität dient dem Gesamttest des Produkts unter dem Aspekt, dass das Produkt geeignet ist, die Geschäftsprozesse in der erwarteten Form zu unterstützen.

Zur Nachweisbarkeit der Tests siehe 3.4.3.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz | x |
| Kompatibilität | x |
| Benutzbarkeit | x |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit | x |
| Portierbarkeit | x |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Systemtests gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der Systemtests in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Systemtests sind verpflichtend. Tailoring ist nur im *Umfang* der Systemtests möglich, muss dokumentiert werden und ist vom Prozessverantwortlichen und von Qualitätsmanagement zu genehmigen.

#### Guidance

Systemtests sind dynamische Tests, bei denen geprüft wird, ob das Gesamtsystem funktioniert. Im Allgemeinen werden End-to-End-Tests durchgeführt. Die Systemtests sollten mindestens so umfangreich wie die Abnahmetests des Kunden sein. Bei geeigneter Planung können dem Kunden die Systemtests auch für die Abnahme zur Verfügung gestellt werden.

#### Abhängigkeiten

Anforderungsspezifikation

Umsetzungskonzept

### Last- und Performancetests

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Last- und Performancetests dienen dazu, das Lastverhalten und Antwort- und Datendurch­satz­ver­hal­ten eines Produkts zu überprüfen und mit den Anforderungen zu vergleichen. Hiermit soll ver­mie­den werden, dass erst im Produktivbetrieb übermäßig lange Antwortzeiten oder inkorrektes oder unerwartetes Verhalten bei hoher Last erkannt werden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität besteht aus den klassischen Schritten des Testmanagements und leitet die Tests aus den Last- und Performance Anforderungen (siehe 3.2.1) ab. Artefakte sind

* Testplan (siehe 3.4.2)
* Testfälle
* Testdaten
* Testnachweise (siehe 3.4.3)

Zielgruppen sind

* Produktmanagement
* Product Owner
* Testteam
* Kunde

Die Artefakte liegen üblicherweise in ausdruckbarer Form (z. B. PDF) und in elektronischer Form (z. B. WIKI) vor.

Der Umfang der Last- und Performancetests wird im Testrahmenplan (siehe 3.4.1) beschrieben und im Testplan (siehe 3.4.2) detailliert. Die zugrundeliegende Konfiguration muss vergleichbar mit Produktionsbedingungen sein.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz | x |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Last- und Performancetests gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der Last- und Performancetests in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Last- und Performancetests sind verpflichtend (außer für SAP). Tailoring für den Entfall der Dokumentation der Testergebnisse ist zu dokumen­tieren und vom Projektverantwortlichen und von Qualitäts­management zu genehmigen. Tailoring für den Entfall von Last- und Performancetests ist zu dokumentieren, zu begründen und vom Projektverantwortlichen und von Qualitätsmanagement zu genehmigen.

#### Guidance

Für Last- und Perfomancetests werden üblicherweise Szenarien ausgewählt, die als kritisch in Bezug auf Lastverhalten und Performanceverhalten eingeschätzt werden. Diese Szenarien sollen – abhängig von den Anforderungen – End-to-End-Szenarien sein. Prozesskenntnisse und Kenntnis der Use Cases ist erforderlich.

Die technische Umsetzung der Messungen erfolgt mit speziellen Tools. Es muss bewusst sein, dass die Nutzung dieser Tools das Verhalten der Systeme beeinflussen kann und ihre Bedienung im Allgemeinen spezielle Qualifikationen erfordert.

Sollten keine Anforderungen in Bezug auf Lastverhalten oder Performanceverhalten vorhanden sein, können derartige Tests auch – unter Verwendung realitätsnaher Szenarien – verwendet werden, um das tatsächliche Last- und Performanceverhalten des Systems herauszufinden. In diesen Fällen ist anhand der Ergebnisse zu bewerten, ob das Verhalten akzeptabel ist und den Erwartungen entspricht.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)
* Testplan (siehe 3.4.2)
* Anforderungsspezifikation (siehe 3.2.1)

### Regressionstests

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Regressionstest dienen dazu, bei Änderungen an einer Applikation das ordnungsgemäße Verhalten der nicht geänderten Teile zu überprüfen. Damit kann das Risiko von Seiteneffekten von Änderungen reduziert werden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität besteht aus den klassischen Schritten des Testmanagements und leitet die Tests aus der Regressionstest-Strategie ab. Artefakte sind

* Regressionstest-Strategie
* Testplan (siehe 3.4.2)
* Testfälle
* Testdaten
* Testnachweise (siehe 3.4.3)

Zielgruppen sind

* Produktmanagement
* Product Owner
* Testteam
* Applikationsmanagement
* Betrieb

Zum Vorliegen der einzelnen Artefakte siehe die jeweiligen Beschreibungen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz (einer Regressionsteststrategie)
3. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz | x |
| Kompatibilität | x |
| Benutzbarkeit | x |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Testnachweise des Regressionstests, kann automatisiert werden

Existenz Regressionsteststrategie:  
Review mit den Kriterien, ob die Testfälle für den Regressionstest gemäß Regressionsteststrategie ausgewählt wurden.

Grad der Testabdeckung: Review der Testnachweise mit den Kriterien  
- Wie viele der Anforderungen wurden durch Tests auf Basis der Regressionsteststrategie geprüft?

#### Guidance

Regressionstests werden üblicherweise automatisiert durchgeführt (siehe ). Als Basis wird eine im Unternehmen verabschiedete Regressionstest-Strategie verwendet, welche die Auswahl der durchzuführenden Testfälle und die Größe der Stichprobe beschreibt.

Zunächst geht man von den bereits automatisierten Testfällen aus. In der Planung des Vorhabens soll jedoch definiert werden, ob weitere automatisierte Testfälle entstehen sollen. Die Kosten müssen im Umsetzungskonzept und Angebot berücksichtigt werden.

#### Abhängigkeiten

* Testautomatisierung (siehe 3.4.4)
* Umsetzungskonzept (siehe 3.2.4)
* Testplan (siehe 3.4.2)

### Sicherheitstests

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Sicherheitstests dienen der Bestätigung, dass die Sicherheitsanforderungen (gesetzlich und selbst definiert) eingehalten sind. Sie sind unerlässlich angesichts der gesetzlichen Regelungen und der Zertifizierung nach ISO 27001.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität besteht aus den klassischen Schritten des Testmanagements und leitet die Tests aus den Anforderungen der Informationssicherheit (siehe 3.2.1) ab. Artefakte sind

* Testplan (siehe 3.4.2)
* Testfälle
* Testdaten
* Testnachweise (siehe 3.4.3)

Zielgruppen sind

* Produktmanagement
* Product Owner
* Testteam
* Betrieb
* Stabsstelle Informationssicherheit
* Kunde

Zum Vorliegen der einzelnen Artefakte siehe die jeweiligen Beschreibungen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit | x |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Sicherheitstests gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der Sicherheitstests in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Sicherheitstests sind verpflichtend. Der Umfang wird im Testrahmenplan, bei kleiner Adaptionen des Produkts im Testplan festgelegt. Tailoring für den Entfall der Dokumentation der Testergebnisse oder der Sicherheitstests ist nicht vorgesehen

#### Guidance

Sicherheitstest dienen dem Nachweis, dass die die Anforderungen aus der Informationssicherheit eingehalten werden. Sie dienen *nicht* der Fehlerfindung. Hierzu gehört der funktionale Test der Anforderungen aus der Informationssicherheit (z. B. Validierung von Eingabefeldern). Methoden sind Unit Tests (siehe 3.4.3), Integrationstests (siehe 3.4.6), funktionale Test sowie Penetrationtests.

Da Sicherheitstests aufwändig sind und spezielle Qualifikationen erfordern (häufig Beauftragung einer externen Spezialfirma), können Sicherheitstests auch auf einer eigenen Zeitschiene geplant werden und in regelmäßigen Abständen (z. B. jährlich) durchgeführt werden. Die Synchronisation dieser Planung mit den geplanten Releases der Produkte muss dann aus inhaltlicher Sicht mit der verantwortlichen Stabsstelle durchgeführt werden.

#### Abhängigkeiten

* Umsetzungskonzept (siehe 3.2.4)
* Systemarchitekturkonzept (siehe 3.2.5)
* Testrahmenplan (siehe 3.4.1)
* Testplan (siehe 3.4.2)

### Verfügbarkeitstests

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Verfügbarkeitstests sind ein Teil der Sicherheitstests und überprüfen die Einhaltung der definierten Verfügbarkeitskriterien (Anforderungen, SLA). Sie stellen die Basis für die Einhaltung der SLA dar.

#### Beschreibung des Themas/Artefakts/der Aktivität

Diese Aktivität besteht aus den klassischen Schritten des Testmanagements und leitet die Tests aus den Verfügbarkeits-Anforderungen (siehe 3.2.1) ab. Artefakte sind

* Testplan (siehe 3.4.2)
* Testfälle
* Testdaten
* Testnachweise (siehe 3.4.3)

Zielgruppen sind

* Produktmanagement
* Product Owner
* Testteam
* Betrieb
* Kunde

Zum Vorliegen der einzelnen Artefakte siehe die jeweiligen Beschreibungen.

Der Umfang der Verfügbarkeitstests wird im Testrahmenplan (siehe 3.4.1) beschrieben und im Testplan (siehe 3.4.2) detailliert. Die zugrundeliegende Konfiguration muss vergleichbar mit Produktionsbedingungen sein.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Verfügbarkeitstests gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der Verfügbarkeitstests in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Verfügbarkeitstests sind verpflichtend. Der Umfang wird im Testrahmenplan, bei kleiner Adaptionen des Produkts im Testplan festgelegt. Tailoring für den Entfall der Dokumentation der Testergebnisse oder der Sicherheitstests ist nicht vorgesehen

#### Guidance

Verfügbarkeitstests betrachten mehrere Aspekte der Verfügbarkeit:

1. „Kurzfristige“ Verfügbarkeit im Betrieb (z. B. Wiederherstellung nach Fehlern, Fehlertoleranz)
2. „Langfristige“ Verfügbarkeit im Betrieb (z. B. Stabilität, erforderliche präventive Wartung)
3. Wiederanlaufverhalten im Katastrophenfall (z. B. vollständiges Restore des Produkts inkl. Daten)

Üblicherweise werden für die Verfügbarkeitstests Szenarien für Ausfälle und Fehler erstellt (z. B. Ausfall einer Datenbank, Ausfall des Netzwerks) und mit Maßnahmen für die Behebung des Fehlers versehen (dies sind dann die Testfälle). In der Folge werden diese vereinbarten Szenarien getestet, wobei die „schwierigen“ Fälle (z. B. Restore des gesamten Rechenzentrums) üblicherweise nicht durchgeführt werden, da die Wahrscheinlichkeit für das Eintreten gering ist, das Risiko bei der Wiederherstellung jedoch ziemlich groß ist.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)
* Testplan (siehe 3.4.2)
* Sicherheitstests (siehe 3.4.10)
* Anforderungsspezifikation (siehe 3.2.1)

### Interner Test in SAP

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der interne Test in SAP dient der Überprüfung, ob alle angeforderten Eigenschaften korrekt umgesetzt sind. Diese Stufe entspricht dem Systemtest für Individualsoftware und ist eine normative/gesetzliche Forderung im Entwicklungsprozess. Mit einer hohen Testabdeckung werden die Risiken für Fehler im Produktivbetrieb reduziert. Insbesondere bei passiver Akzeptanz durch den Kunden ist dies die wesentliche abschließende Teststufe.

#### Beschreibung des Themas/Artefakts/der Aktivität

Der interne Test in SAP ist der Systemtest der ITSV, der zur Erklärung einer Abnahmebereitschaft oder zu Erklärung der Produktivsetzungsbereitschaft auf ITSV-Ebene erforderlich ist. Hier wird nachgewiesen, dass das System im Rahmen der definierten Testszenarien gemäß den Anforderungen (Verifikation) funktioniert. Das positive Ergebnis des internen Tests in SAP ist der Schritt, den externen Test in SAP beim Kunden zu starten und dadurch die Abnahme und Produktivsetzung zu erreichen.

Für SAP gibt es keinen Unittest, Integrationstest und Systemtest. Diese Teststufen werden durch den internen Test in SAP abgelöst.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Grad der Testabdeckung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | x |
| Effizienz | x |
| Kompatibilität |  |
| Benutzbarkeit | x |
| Zuverlässigkeit | x |
| Sicherheit | x |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der internen Tests in SAP gemäß definierter Vorgaben, kann teilweise automatisiert werden.

Grad der Testabdeckung: Review der Dokumentation der internen Tests in SAP in Bezug auf den vorgegebenen Grad der Testabdeckung.

*Tailoring*: Kein Tailoring vorgesehen.

#### Guidance

In standardisierten Umfeldern wie SAP ist es sinnvoll, die Tests zu automatisieren und entweder alle automatisiert durchzuführen (Regressionstests, siehe ). Testfälle für neue Funktionalität werden sinnvoller Weise nach Erstellung automatisiert und in den Pool der Testfälle aufgenommen. Werden nicht alle Testfälle automatisiert durchgeführt oder sind einzelne Bereich der Tests nicht automatisiert oder automatisierbar, sind manuelle Tests gemäß den klassischen Testmanagementmethoden auszuwählen und dokumentiert durchzuführen.

#### Abhängigkeiten

* Testrahmenplan (siehe 3.4.1)
* Testplan (siehe 3.4.2)
* Regressionstest (siehe 3.4.9)

### Externer Test in SAP (durch Träger)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der Externe Test in SAP entspricht dem Abnahmetest. Mit der erfolgreichen Durchführung wird nachgewiesen, dass das Vorhaben auf der Umgebung des Kunden funktioniert und damit die Abnahme ausgesprochen wird.

#### Beschreibung des Themas/Artefakts/der Aktivität

Der externe Test in SAP (durch Kunden) ist der Abnahmetest eines Vorhabens, mit dem über die Abnahme und die Produktivsetzung entschieden wird. Der externe Test liegt in der Verantwortung des Kunden. Das Ergebnis der externen Tests in SAP sind idealerweise

* Fehlertickets (falls Fehler auftreten)
* Abnahmeprotokoll

Fehlertickets liegen üblicherweise in elektronischer Form in einem Tool vor. Das Abnahmeprotokoll liegt üblicherweise in ausdruckbarer Form unterschrieben vor.

Der Externe Test in SAP löst den Akzeptanztest (in der Verantwortung des Kunden) ab.

#### Beschreibung des Q-Kriteriums

1. Existenz

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung | X |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit | X |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation der Abnahme des Kunden. Diese Dokumentation muss ITSV-intern erstellt werden, wenn die Frist für die Erklärung der Abnahme seitens des Kunden verstrichen ist.

*Tailoring*: Kein Tailoring vorgesehen.

#### Guidance

Die Verantwortung für den externen Test in SAP liegt beim Kunden. Dies bedeutet, dass der Kunde die Testfälle auswählt und durchführt, die aus seiner Sicht zur Abnahme und für die Produktivsetzung erforderlich sind. Hierbei kann er selbstverständlich von der ITSV unterstützt werden, sei es in Bezug auf Testfallauswahl, Testfalldefinition oder Testdurchführung. Sollte passive Akzeptanz bei Nichtreaktion des Kunden innerhalb einer definierten Zeitspanne vorausgesetzt werden, so ist die Bestätigung der passiven Akzeptanz nach Überschreitung der Frist sinnvoll, um einen Nachweis für die Abnahme zu haben.

#### Abhängigkeiten

* Interner Test in SAP (siehe 3.4.12)

### Akzeptanztest

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Der Akzeptanztest dient der Abnahme des Produkts und stellt den Verantwortungsübergang in die nächste Phase dar.

#### Beschreibung des Themas/Artefakts/der Aktivität

Der Akzeptanztest erfolgt in der Verantwortung des Kunden. Die Rolle der ITSV beschränkt sich auf Bereitstellung des Systems mit dem abzunehmenden Produkt, im Falle der Mandanten-TA2-Systeme auch nur auf das Bereitstellen des Produkts über RDM.

#### Beschreibung des Q-Kriteriums

1. Existenz

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Bestätigung des Kunden über eine erfolgreiche Abnahme Diese Dokumentation muss ITSV-intern erstellt werden, wenn die Frist für die Erklärung der Abnahme seitens des Kunden verstrichen ist.

*Tailoring*: Kein Tailoring vorgesehen.

#### Guidance

Die Verantwortung für den Akzetanztest liegt beim Kunden. Dies bedeutet, dass der Kunde die Testfälle auswählt und durchführt, die aus seiner Sicht zur Abnahme und für die Produktivsetzung erforderlich sind. Hierbei kann er selbstverständlich von der ITSV unterstützt werden, sei es in Bezug auf Testfallauswahl, Testfalldefinition oder Testdurchführung. Sollte passive Akzeptanz bei Nichtreaktion des Kunden innerhalb einer definierten Zeitspanne vorausgesetzt werden, so ist die Bestätigung der passiven Akzeptanz nach Überschreitung der Frist sinnvoll, um einen Nachweis für die Abnahme zu haben.

#### Abhängigkeiten

* Systemtest (siehe 3.4.7)

## Inbetriebnahme

### Dokumentation neuer/geänderter Parameter

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die Dokumentation geänderter Parameter unterstützt die Inbetriebnahme und vermeidet falsche oder inkonsistente Einstellungen im operativen Betrieb.

#### Beschreibung des Themas/Artefakts/der Aktivität

Die Dokumentation neuer/geänderter Parameter beschreibt, was sich innerhalb einer Release in Bezug auf die Parameter und ihre Belegung geändert hat bzw. welche Parameter neu sind und nun definiert werden müssen.

Zielgruppen sind

* Integrationsteam
* Testteam
* Betrieb
* Applikationsmanagement

Die Dokumentation neuer/geänderter Paramater liegt im Allgemeinen in ausdruckbarer Form vor. Sie kann auch Bestandteil der Release Notes (siehe 3.3.5) sein.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Vollständigkeit
3. Inhaltliche Korrektheit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | X |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Dokumentation (unabhängig von Medium), kann (teilweise) automatisiert werden

Vollständigkeit: Review mit den Kriterien  
- Sind alle Parameter mit dem alten und dem neuen Wert beschrieben?

Inhaltliche Korrektheit: Review mit den Kriterien  
- Sind die angegebenen Werte korrekt?  
- Stimmen die angegebenen Werte mit der Entwicklungsdokumentation überein?

#### Guidance

Bei Änderungen von Parametereinstellungen zwischen Releases oder bei Änderungen im Vergleich zum vom Hersteller gesetzten Defaultwert ist die Dokumentation für das Installationsteam essentiell. Die Dokumentation sollte alle geänderten Parameter enthalten. Sollte eine gesonderte Dokumentation aller Parameter bestehen, kann diese Dokumentation auf den aktuellen Stand referenzieren. Existieren beide Dokumentationen muss auf die Konsistenz geachtet werden.

Dieses Artefakt kann Bestandteil der Release Notes sein.

#### Abhängigkeiten

* Release Notes (siehe 3.3.5)

### Einordnung in den Service Katalog des Rechenzentrums

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Die Zuordnung von Applikationen zum Katalog des Service Katalog des Rechenzentrums erleichtert einerseits die interne Kommunikation und die Bereitstellung der erforderlichen Maßnahmen im Rechenzentrum, andererseits die Definition von SLAs mit dem Kunden und das dazugehörige Service Management (ITIL).

#### Beschreibung des Themas/Artefakts/der Aktivität

Dieses Thema legt fest, welche Services aus dem Service Katalog des Rechenzentrums für den Betrieb einer Applikation zutreffend sind. Diese Einordnung muss mit den Anforderungen an das SLA abgeglichen werden.

Zielgruppen sind

* Service Management
* Produktmanagement
* Controlling
* Verrechnung

Üblicherweise liegt diese Information anfänglich in einem eigenen Dokument vor oder ist Bestandteil eines anderen Dokuments (z. B. System Architektur Konzept). In der Folge, spätestens bei Übergabe an den Betrieb wird die Information in die Dokumentation der Services des Rechenzentrums und des Produktmanagements der Applikation aufgenommen.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Korrektheit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | X |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Zuordnung

Korrektheit: Prüfung der Zuordnung in Bezug auf bestehende Qualitätsanforderungen und Vereinbarungen im Rechenzentrum und der Applikation

#### Guidance

Der Betrieb von Produkten wir üblicherweise mehrere Services des Rechenzentrums verwenden. Daher ist es bereits in einer sehr frühen Phase (z. B. Umsetzungskonzept) sinnvoll, die ersten Über­legungen zu diesem Thema anzustellen, insbesondere, wenn neue Services oder geänderte Leis­tungsmerkmale der Services zu erwarten sind. Mit dieser Zuordnung können die SLAs verein­heit­licht werden.

#### Abhängigkeiten

* Release Notes (siehe 3.3.5)
* Umsetzungskonzept (siehe 3.2.4)
* SLA (siehe 3.6.1)
* Anforderungsspezifikation (siehe 3.2.1)

### Installationsanleitung

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Installationsanleitungen sind erforderlich, die gelieferten Applikationen korrekt für den Produktiv­betrieb zu installieren. Mit Installationsanleitungen kann vermieden werden, Applikationen unvoll­stän­dig oder fehlerhaft zu installieren und dadurch Verzögerungen oder Fehler hervorzurufen. Weiters sind Installationsanleitungen für den Desaster Recovery Fall erforderlich.

#### Beschreibung des Themas/Artefakts/der Aktivität

Die Installationsanleitung enthält die Beschreibung der einzelnen Schritte, die für die Installation und Inbetriebnahme der Applikation erforderlich.

Zielgruppen sind

* Betrieb des Kunden
* Applikationsmanagement
* Betrieb

Die Installationsanleitung liegt üblicherweise in revisionssicherer, ausdruckbarer Form (z. B. PDF) vor. Alternativ kann sie – insbesondere bei gleichbleibenden Schritten – in elektronischer Form (z. B. WIKI) vorliegen. Hierbei muss jedoch die Verwendung im Katastrophenfall berücksichtigt werden.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Inhaltliche Korrektheit
3. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)
* Inhaltliches Review (für C.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der PDF-Datei, kann automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Zielgruppenorientierung der Beschreibung  
- Korrektheit der Beschreibungen für die Installation

Vollständigkeit: Review mit den Kriterien  
- Sind alle Schritte beschreiben?  
- Sind die Abhängigkeiten bei der Installation beschreiben?  
- Zielgruppenorientierung der Beschreibung

#### Guidance

Installationsanleitungen sind essentiell für die Übergabe des Entwicklungsteams an das Applikationsmanagement und den Betrieb. Die Unterlagen müssen inhaltlich korrekt sein und den Wissenstand der Durchführenden berücksichtigen. Bei vollständigen Installationsanleitungen müssen die Schritte so beschreiben sein, dass auch minder erfahrene Personen diese Aktivitäten durchführen können, ohne die Gepflogenheiten der ITSV im Detail zu kennen.

#### Abhängigkeiten

* Release Notes (siehe 3.3.5)
* Umsetzungskonzept (siehe 3.2.4)

### Ablage der Pakete und Konfigurationsdaten für die Installation

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Für eine vollständige und korrekte Inbetriebsetzung müssen alle aktuellen Pakete und Konfigura­tions­daten für die Installation vorhanden sein. Gibt es an dieser Stelle Fehler, kann in der Folge der Betrieb gefährdet sein, Installationen sind verspätet oder es gibt im schlimmsten Fall Fehler zu einem nicht definierten Zeitpunkt (bei selten genutzten Funktionen), die nur schwer zu finden sind.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Thema der Ablage der Pakete und Konfigurationsdaten für die Installation ist ein Detailthema des Konfigurationsmanagements. Es geht darum, dass der Übergabepunkt von Entwicklung an den Betrieb eindeutig definiert ist und alle erforderlichen Pakete für eine Installation korrekt abgelegt sind. Inhalte sind

* Liste aller erforderliche Pakete
* Aktuelle Pakete
* Zugehörige Konfigurationsdaten

Zielgruppen sind

* Entwicklung
* Applikationsmanagement
* Betrieb
* Betrieb des Kunden

Die Liste aller erforderlichen Pakete liegt im Allgemeinen in ausdruckbarer Form (z. B. PDF) vor.

#### Beschreibung des Q-Kriteriums

1. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Formales Review (für A.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Vollständigkeit: Sichtung der gelieferten Pakete (automatisierbar)  
- Vergleich mit der Liste der Soll-Pakete  
- Überwachung der Vollständigkeit  
- Überprüfung der Konfigurationsdaten auf Vollständigkeit

*Tailoring*: Es kann (nachweislich) vereinbart werden, dass nur geänderte Konfigurationsdaten geliefert werden.

#### Guidance

Die Übereinstimmung von Lieferliste und tatsächlich geliefertem Umfang garantiert nicht die Voll­stän­digkeit, erleichtert aber in der Folge eine Fehlerfindung und erhöht das Bewusstsein bei der Aus­wahl.

Nichtzusammengehörige oder fehlende Pakete werden im Idealfall bei der Installation identifiziert. Gefährlich sind Pakete, die auf den „ersten Blick“ passen, die aber veraltet sind und neu definierte (seltene) Fälle der Funktionalität nicht abdecken. Dies kann von einem Fehlerverhalten bis zum Absturz der Applikation führen.

Ebenso wichtig wie die Pakete sind die Konfigurationsdaten („Parametrierung“). Auch wenn die meisten Parameter Defaultwerte besitzen, ist es sinnvoll, diese zu dokumentieren und vollständig zu übergeben, da nicht sichergestellt ist, dass sich die Defaultwerte von einer Version des Produkts auf die nächste nicht ändern oder bei Änderungen und Adaptionen die Werte nicht korrekt dokumentiert wurden. Mit der Ablage des vollständigen Satzes an Konfigurationsdaten wird eine dokumentierte Baseline erstellt.

Dieser Punkt adressiert einen durchgängigen Prozess zum Konfigurationsmanagement und muss dort konzeptionell verankert werden.

Die abgelegten Pakete müssen mit der Liste (z. B. in der Release Note) übereinstimmen.

Eine Überprüfung der Korrektheit der Konfigurationsdaten an dieser Stelle ist aufwändig.

* Release Notes (siehe 3.3.5)
* Software Design (siehe 3.3.1)

### Laufzeitverhalten im Vergleich mit vorheriger Release

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Sollte sich das Laufzeitverhalten eines Produkts zwischen zwei Releases grundlegend verändern, sollten insbesondere bei Laufzeitverlängerungen die Benutzer auf diese Tatsache rechtzeitig hingewiesen werden und eine Begründung gegeben werden. Dieser unabhängige Test auf einer produktionsnahmen Umgebung ergänzt den Last- und Performancetest.

#### Beschreibung des Themas/Artefakts/der Aktivität

Dieses Thema behandelt das Laufzeitverhalten (Performance) zwischen zwei aufeinanderfolgenden Releases. Implizite Anforderung ist, dass sich das Laufzeitverhalten nicht „signifikant“ ändert. Hierzu werden Messungen von End-to-End-Szenarien durchgeführt und dokumentiert. Artefakte sind

* Testfälle
* Testdaten
* Testnachweise (siehe 3.4.3)

Zielgruppen sind

* Produktmanagement
* Product Owner
* Applikationsmanagement
* Testteam
* Kunde

Die Ergebnisse liegen üblicherweise als Protokoll der Messung in ausdruckbarer Form (z. B. PDF) vor. Für die anderen Artefakte siehe 3.4.7.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Qualitative Übereinstimmung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz | x |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) der Entscheidung (für A.)
* Vergleich der Laufzeiten mit den Laufzeiten der letzten Release(s) (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Existenzprüfung der Entscheidung (Sichtkontrolle) (für A.)

Vergleich der Ergebnisse: Bewertung der Laufzeitdifferenz für die definierten Testfälle

*Tailoring*: Es können für die vorgegebene erlaubte Maximalabweichung zusätzliche Toleranzgrenzen angegeben werden.  
Wenn bereits ein Last- und Performancetest erfolgreich war, kann dieser Schritt entfallen.

#### Guidance

Für diese Aktivität gibt es unterschiedliche Ausprägungen, z. B. Test im Sinne eines Last- und Performancetests unter Benutzung eines Tools, Messung einzelner komplexer Schritte, Messung eines End-to-End-Durchlaufs eines Prozesses. Definition des Testfalls, der zugehörigen Testdaten und der Messmethode ist jedenfalls erforderlich.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Software Design (siehe 3.3.1)
* Last- und Performancetests (siehe 3.4.8)

## Betrieb

### Service Level Agreement (SLA)

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Mit einem SLA werden die Anforderungen an die Qualität des Service definiert. Die Definition (oder Adaption) eines SLA vor Produktivsetzung hilft, die Aktivitäten des Betriebs sinnvoll zu planen und spätere Diskussionen über nicht erfüllte Erwartungen des Kunden zu vermeiden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Artefakt SLA definiert die messbaren nichtfunktionalen Anforderungen, welche im Betrieb eingehalten werden müssen. Die Ergebnisse der Einhaltung werden regelmäßig an den Kunden berichtet und sind üblicherweise bei Abweichungen (Nichteinhaltung) pönalepflichtig. Zumindest muss die Nichteinhaltung nachvollziehbar begründet und dokumentiert werden.

Zielgruppen sind

* Kunde
* Betrieb
* Produktmanagement

Üblicherweise stehen SLAs in ausdruckbarer Form zur Verfügung (Vertragsbestandteil).

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Übereinstimmung mit Anforderungen und Erwartungen
4. Inhaltliche Korrektheit
5. Nachweis der Einhaltung

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Formales Review (für B.)
* Inhaltliches Review (für C.)
* Inhaltliches Review (für D.)
* Überprüfung der Nachweise (im SLA-Reporting) (für E.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein der Information in einer revisionssicheren Form, kann automatisiert werden.

Existenz Mindestinhalt: Review der ausgefüllten Struktur gemäß den definierten Minimalanforderungen, kann automatisiert werden.

Erwartungen: SLA wird zwischen Kunde und Dienstleister abgeschlossen (Vertrag)

Inhaltliche Korrektheit: Review mit den Kriterien   
- Sind die relevanten Anforderungen im SLA enthalten?  
- Sind die die definierten Anforderungen umsetzbar?

Einhaltung: Prüfung auf Einhaltung der Grenzwerte in den monatlichen Reports

*Tailoring*: SLAs sollten in jedem Fall erstellt werden. Wünscht der Kunde kein SLA sollte das Produktmanagement Vorgaben aus der Erfahrung machen, welche die Erwartungen des Kunden abdecken.

#### Guidance

Die Messkriterien der SLAs werden häufig in Reportingtools hinterlegt, um automatisch die Berichte über die (üblicherweise monatliche) Erfüllung erstellen zu können. Weiters werden die Messkriterien Trendanalysen im Betrieb zugrunde gelegt, um rechtzeitig die potentielle Nichteinhaltung vermeiden zu können.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)

### Betriebshandbuch

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Das Betriebshandbuch soll alle für den Betrieb erforderlichen Aktivitäten beschreiben und Informa­tionen bereitstellen, welche den routinemäßigen Betrieb eines Produkts ohne zusätzliche Kenntnis ermög­licht (1st und 2nd Level Support ohne 3rd Level Support).. Damit wird einerseits die Aufrechterhaltung der Betriebsfähigkeit auch für Personen, die mit dem Produkt nicht vertraut sind, andererseits die Einschulung neuer Mitarbeiter oder die Möglich­keit des Nachlesens in kritischen oder selten auftretenden Fällen ermöglicht.

#### Beschreibung des Themas/Artefakts/der Aktivität

Das Betriebshandbuch beschreibt die für den Betrieb einer Applikation bzw. einer Gruppe von Applikationen erforderlichen Maßnahmen, die im Rechenzentrum getroffen werden müssen. Üblicher­weise sind enthalten

* Verweise auf die Installationsanleitung, auf Anleitungen zum Applikationsmanagement (siehe ), Umsetzungskonzept (siehe 3.2.4)
* Betriebsprozesse bzw. - falls anderweitig zentral dokumentiert – Spezifika, die für die einzelnen Applikationen beachtet werden müssen.
* Wechselwirkungen von Applikationen
* Liste der Applikationen

Zielgruppe sind

* Betrieb
* Betrieb des Kunden

Das Betriebshandbuch steht üblicherweise in ausdruckbarer Form (z. B. PDF) zur Verfügung. In manchen Fällen steht es in elektronischer Form zur Verfügung.

#### Beschreibung des Q-Kriteriums

1. Existenz
2. Existenz Mindestinhalt
3. Inhaltliche Korrektheit
4. Vollständigkeit

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Existenzprüfung (Sichtkontrolle) (für A.)
* Inhaltliches Review (für B.)
* Inhaltliches Review (für C.)
* Formales/inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Existenz: Vorhandensein des Betriebshandbuchs in unveränderbarer Form, kann (teilweise) automatisiert werden

Existenz Mindestinhalt: Review der ausgefüllten Struktur gemäß Vorlage (TBD)

Inhaltliche Korrektheit: Review mit den Kriterien  
- Zielgruppenorientierung der Beschreibung  
- Korrektheit der enthaltenen Informationen

Vollständigkeit: Review mit den Kriterien  
- Sind alle erforderlichen Informationen vorhanden?  
- Zielgruppenorientierung der Beschreibung

#### Guidance

Das Betriebshandbuch ist das zentrale Nachschlagewerk für den Rechenzentrumsbetrieb für die Aufrechterhaltung der Services. Das Betriebshandbuch enthält alle erforderlichen Informationen, die im Regelbetrieb und in Sonder- oder Notfällen erforderlich sind. Hierzu gehören auch empfohlene Maßnahmen bei Auftreten von Fehlern, für den Katastrophenfall sowie Hinweise auf unbedingt erforderliche Rahmenbedingungen (z. B. Speicherplatz, Bandbreiten, Platten­geschwin­dig­keit) als Mindestanforderungen.

Das Betriebshandbuch soll von den Verantwortlichen für die Erstellung des Produkts geschrieben und vom Betrieb abgenommen werden. Wesentlicher Punkt ist hier der Know-how-Transfer.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)

### Application Monitoring

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Application Monitoring dient dazu, schnell zu erkennen, ob ein Produkt (eine Applikation) ord­nungs­gemäß arbeitet. Rechtzeitiges Erkennen von Fehlfunktionen und Problemen erlaubt eine schnelle Reaktion, um fehlerhafte Ergebnisse bis hin zur Nichtverfügbarkeit einer Applikation nach Möglich­keit zu reduzieren und zu vermeiden.

#### Beschreibung des Themas/Artefakts/der Aktivität

Application Monitoring ist eine Aktivität, die den ordnungsgemäßen Betrieb des Produkts über­wacht. Dabei gibt es einen fachlichen Teil, bei dem man überwacht, ob das Produkt inhaltlich kor­rekt arbeitet und einen technischen Teil, bei dem man überwacht, ob das Produkt technisch verfüg­bar ist und technisch korrekt arbeitet.

Durchführende sind das Administrationspersonal, das entsprechende fachliche Kenntnisse hat.

Grundlage ist eine Anleitung zum Application Monitoring, Diese liegt im Allgemeinen in elektronischer oder in ausdruckbarer Form vor. Beispiele sind PDF oder WIKI.

#### Beschreibung des Q-Kriteriums

1. Qualifikation des durchführenden Personals
2. Existenz (einer Anleitung zum Application Monitoring)
3. Inhaltliche Korrektheit (der Anleitung zum Application Monitoring)
4. Vollständigkeit (der Anleitung zum Application Monitoring)

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit |  |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges | x |

#### Arten der Überprüfung des Q-Kriteriums

* Ausbildungsnachweis des durchführenden Personals (für A.)
* Existenzprüfung (Sichtkontrolle) (für B.)
* Inhaltliches Review (für C.)
* Inhaltliches Review (für D.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Qualifikation: Das durchführende Personal hat ausreichende Kenntnis der Funktionen und Wirkungsweise des Produkts, um technisches oder fachliches Fehl­ver­halten des Produkts schnell erkennen und korrigierende Maßnahmen set­zen zu können.

Existenz: Vorhandensein der Anleitung, kann automatisiert werden

Inhaltliche Korrektheit: Review mit den Kriterien  
- Zielgruppenorientierung der Beschreibung  
- Korrektheit der Fehlermöglichkeiten und der Maßnahmen

Vollständigkeit: Review mit den Kriterien  
- Sind alle Fehlermöglichkeiten beschrieben?  
- Sind zu allen Fehlermöglichkeiten Maßnahmen beschrieben  
- Zielgruppenorientierung der Beschreibung

#### Guidance

Application Monitoring erfolgt in unterschiedlichen Ebenen. Das Monitoring der Infrastruktur kann üblicherweise über Standard-Tools eines Rechenzentrums erfolgen. Das Monitoring der Applikationen hat zwei Ebenen:

1. Ist die Applikation „am Leben“? Dies kann durch regelmäßige Outputs dargestellt werden, muss aber bereits in der Entwicklung vorgesehen werden.
2. Arbeitet die Applikation korrekt? Dies kann üblicherweise nur schwer ohne Fachkenntnisse verfolgt werden. Hier müssten inhaltliche Prüfungen in den Applikationen eingebaut werden (Plausibilitätschecks), deren Ergebnisse im Application Monitoring angezeigt werden.

Wird dediziertes qualifiziertes Personal für das Application Monitoring eingesetzt, reduziert sich – abhängig von den Kenntnissen – die Dringlichkeit der Dokumentation. In jedem Fall muss das Personal in der Lage sein, Fehlverhalten schnell zu erkennen und Maßnahmen einleiten zu können (ggf. auch Alarmierung von Spezialisten).

Ein übergeordnetes Konzept für alle Applikationen erscheint wichtig und sinnvoll.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)
* Design for Testability (siehe 3.2.6)

### Gesundheitscheck Webservices

#### Zweck und Nutzen des Themas/Artefakts/der Aktivität

Produkte greifen auf Webservices zu. Damit die Funktionalität von Produkten sichergestellt ist und die Kopplung des Produkts mit anderen Teilen des Systems gewährleistet ist, müssen die zugehörigen Webservices verfügbar sein und funktionieren.

Bei jeder Release wird geprüft, ob die Webservices „am Leben sind“, d. h. ob sie verfügbar sind und die Basisfunktionalität gegeben ist.

#### Beschreibung des Themas/Artefakts/der Aktivität

Dies Testaktivität überprüft, ob ein Webservice korrekt arbeitet. Diese wiederkehrende Aktivität im Betrieb erfolgt im Allgemeinen automatisch durch Scripts. Die Ergebnisse werden im Application Monitoring der zugehörigen Applikation oder des Webservice angezeigt.

#### Beschreibung des Q-Kriteriums

1. Verfügbarkeit
2. Korrekte Funktionalität

#### Einordnung nach ISO 25010

|  |  |
| --- | --- |
| Funktionale Eignung |  |
| Effizienz |  |
| Kompatibilität |  |
| Benutzbarkeit |  |
| Zuverlässigkeit | x |
| Sicherheit |  |
| Wartbarkeit |  |
| Portierbarkeit |  |
| Sonstiges |  |

#### Arten der Überprüfung des Q-Kriteriums

* Test der Webservices auf Verfügbarkeit (für A.)
* Test der Webservices auf Funktionalität (für B.)

#### Bedingungen, wann das Q-Kriterium erfüllt ist

Verfügbarkeit: Antwort des Webservice entsprechend den definierten Testfällen

Funktionalität: Korrekte Antwort des Webservice entsprechend den definierten Testfällen

#### Guidance

Generell sollte davon ausgegangen werden, dass Webservices, die nicht geändert wurden, korrekt funktionieren. Daher ist es üblicherweise nur erforderlich, die Verfügbarkeit mit definierten Testfällen zu überprüfen.

Wurden jedoch im Zuge der Änderung der Applikation Änderungen an den Schnittstellen und an der Funktionalität der Webservices durchgeführt, ist ein Integrationstest erforderlich bzw. muss die Basisfunktionalität geprüft werden (außerhalb dieses Punktes).

Als zusätzlicher Gesichtspunkt kommt hier der Umfang des Tests ins Spiel. Beim Test auf Verfügbarkeit können ausgewählte oder alle Webservices geprüft werden. Wenn nicht alle Webservice geprüft werden, sollte die Auswahl risikobasiert erfolgen, wobei die Auswahlkriterien „Kritikalität des Webservice“ und „Anzahl der Webserviceaufrufe“ sein können. Beim Test auf Funktionalität gelten alle üblichen Regeln für einen Regressionstest bzgl. getestetem Funktionsumfang. Für die Auswahl der zu testenden Webservices gelten die Überlegungen zur Prüfung der Verfügbarkeit der Webservices.

#### Abhängigkeiten

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)
* Design for Testability (siehe 3.2.6)

## Applikationsbetrieb managen

Abgedeckt durch

* Application Monitoring (siehe 3.6.3)
* SLA (siehe 3.6.1)

## Sunset

Abgedeckt durch

* Anforderungsspezifikation (siehe 3.2.1)
* Umsetzungskonzept (siehe 3.2.4)

## Sonstiges Objekte

### Softwareprodukt

Ein Softwareprodukt ist ein mittels der ITSV DevOps Toolchain installierbares Software-Paket, das die Erfüllung sämtlicher Anforderungen der dem betrachteten Entwicklungsprojekt zugrundeliegenden Anforderungsspezifikation beinhaltet.

Die Kriterien für ein Softwareprodukt werden durch den Prozess mit seinen Quality Gates, die Richtlinien und die Bewertung des Softwareprodukts im Testprozess, insbesondere im Akzepanztest definiert.

Ein in Betrieb befindliches Software Produkt ist synonym zu abgenommener, deployter, laufender Software.

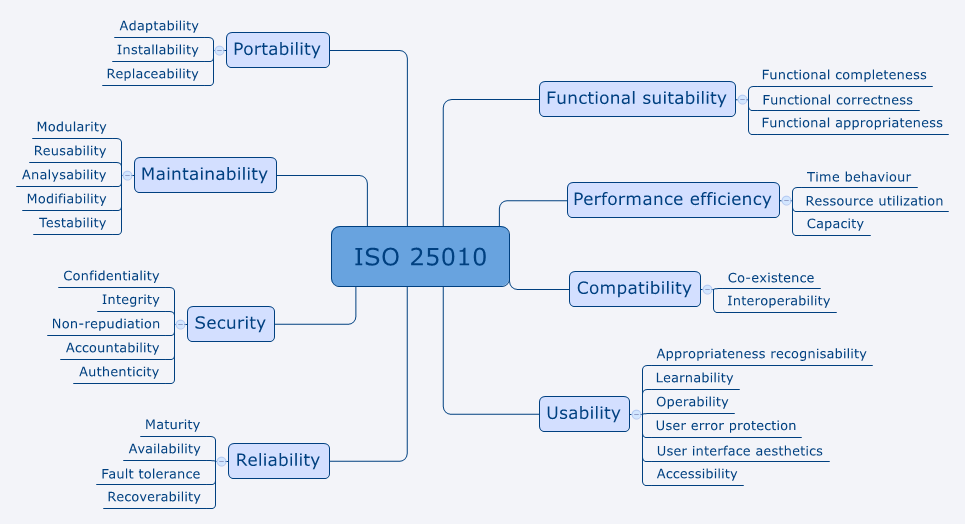
# Offene Punkte

|  |  |
| --- | --- |
| **Titel** | **Offene Frage** |
| Kriterien des CAB | Welches sind die Kriterien für die Genehmigung eines CR im CAB? |
| Teamqualität in agilen Projekten | Agile Teams „leben“ von der Qualität der Teams. Gute agile Vorgehensweisen erfordern eingespielte Teams. Dies bedeutet, dass es auf emotionaler/sozialer Ebene Kriterien für die Teamqualität gibt. Diese wurden in diesem SWQHB nicht berücksichtigt, sind aber bei konsequenter Umsetzung von agilen Methoden dringend erforderlich. |
| Terminologie | System oder Applikation oder Produkt (vereinheitlichen) |
| Tailoring | Wer genehmigt Abweichungen von den Vorgaben? |

# Anhang A: Checklisten und Protokolle

## Detaillierung ISO 25010

Die ISO 25010 sieht acht Qualitätskriterien für ein Software-Produkt, die jeweils wieder detailliert werden.



## Vorlage Release Notes

## Checkliste Release Notes

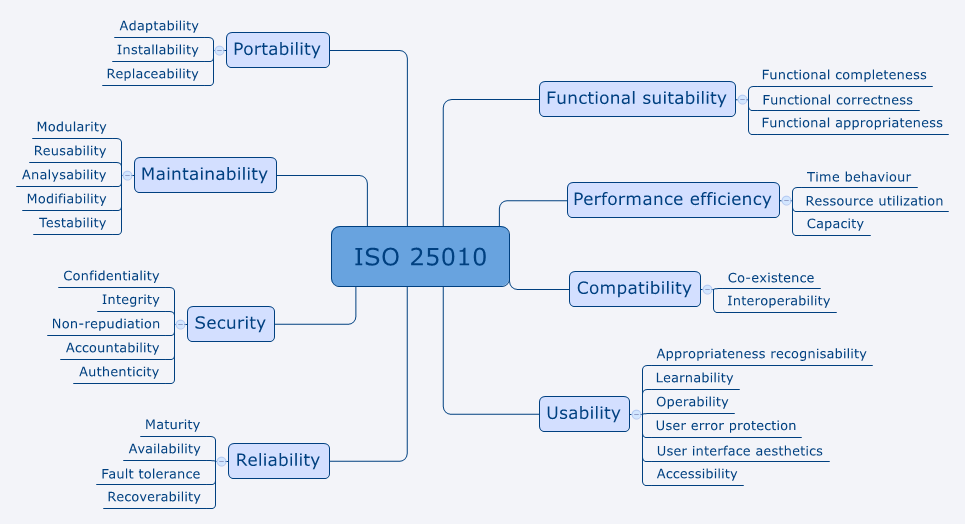
## Checkliste Anforderungsspezifikation

Folgende Fragen sollen positiv beantwortet werden:

* Sind die Anforderungen dokumentiert?
* Sind die Anforderungen verbindlich mit dem Kunden abgestimmt?
* Sind bei Änderungen die Auswirkungen auf das Gesamtsystem analysiert worden?
* Sind die Anforderungen aus Sicht der Stakeholder vollständig?  
  - funktional  
  - nicht-funktional
* Ist das Gesamtsystem eindeutig abgegrenzt?
* Sind die Schnittstellen eindeutig und vollständig festgelegt?  
  - intern  
  - extern
* Bei Teilsystemen: Ist die Einbettung des Teilsystems in das Gesamtsystem eindeutig festgelegt?
* Ist eindeutig festgelegt, wo die Grenze der Verantwortung ist?
* Sind die nicht funktionalen Anforderungen ausreichend?  
  - Benutzbarkeit  
  - Performance  
  - Wartbarkeit  
  - Zuverlässigkeit
* Ist die Erwartungshaltung des Kunden klar?
* Haben Kunde und ITSV GmbH ein gleiches Verständnis der Anforderungen?
* Sind neben den Kundenanforderungen auch die internen Vorgaben in den Anforderungen berücksichtigt worden?
* Sind die Anforderungen verständlich formuliert?
* Ist die Terminologie der Formulierung einheitlich?
* Ist Gleiches gleich benannt?
* Sind Fachwörter verständlich?
* Beschreiben die funktionalen Anforderungen auch den Ablauf?
* Sind die nicht-funktionalen Anforderungen messbar definiert?
* Sind die Anforderungen eindeutig formuliert, so dass sie testbar sind?
* Kann man aus den Anforderungen ableiten, wie sie mit vertretbarem wirtschaftlichen Aufwand getestet werden können?
* Kann man aus den Anforderungen erkennen, welche Zusatzinformation zur Definition eines Testfalls noch erforderlich ist?
* Sind die Anforderungen redundanzfrei?
* Sind die Abhängigkeiten der einzelnen Anforderungen untereinander klar?
* Gibt es eine Dokumentation der Abhängigkeiten der einzelnen Anforderungen voneinander?  
  - auf gleicher Abstraktionsebene  
  - auf unterschiedlicher Abstraktionsebene (Verfeinerungen)
* Ist das Risiko der einzelnen Anforderungen bekannt?  
  - Was bedeutet es, wenn die Anforderung nicht erfüllt ist?  
  - Was bedeutet es, wenn die Anforderung falsch umgesetzt wird?  
  - Was bedeutet es, wenn die Anforderung temporär nicht verfügbar ist?
* Ist die Gesamtheit der Anforderungen umsetzbar:  
  - technisch  
  - mit den vorhandenen Qualifikationen  
  - mit den verfügbaren Ressourcen  
  - mit dem geplanten Budget
* Welchen Einfluss kann der Kunde auf den Erfolg des Vorhabens haben?
* Sind die Anforderungen an die Informationssicherheit adäquat berücksichtigt worden?
* Wurde bei den einzelnen Anforderungen überlegt, welche Auswirkungen diese Anforderung auf die Informationssicherheit hat?
* Sind die Anforderungen formal freigegeben?  
  - intern- vom Kunden
* Unterliegen die Anforderungen einem definierten Änderungsprozess?
* Sind die Anforderungen für alle Betroffenen verfügbar?
* Ist sichergestellt, dass alle Betroffenen mit den aktuellen Anforderungen und nicht mit veralteten arbeiten?

## Inhalte Anforderungsspezifikation

Sind alle Punkte der ISO 25010 adressiert (ggf. auch als nicht anwendbar)?



Spezielle Anforderungsthemen der ITSV GmbH:

* Randbedingungen wie
  + Release Schein
  + Upgradepfade
  + Coding Rules
  + Einstufung des Schutzbedürfnisses (Datenschutzbestimmungen)
  + Richtlinien der Software Entwicklung
* Explizite funktionale Anforderungen
* Implizite funktionale Anforderungen
* Abgeleitete funktionale Anforderungen
* Explizite nicht-funktionale Anforderungen
* Implizite nicht-funktionale Anforderungen
* Abgeleitete nicht-funktionale Anforderungen
* Anforderungen des Testteams an das Design for Testability
* Anforderungen aus einem vorhergehenden System
  + Verbesserungsvorschläge
  + Ablöse von Systemen (Sunset)
* Anforderungen aus Sicht des Application Monitoring
* Anforderungen an Schnittstellen zu Webservices
* Anforderungen an die Anbindung der Alamierung
* Anforderungen aus Sicht des Monitorings des Rechenzentrums
* Anforderungen aus dem Vertrag mit dem Kunden
* Gesetzliche Vorgaben
* Design Patterns in der Entwicklung
* Layerstruktur der Software
* Automatische Generierung von Software
* Historisierung
* Archivierung
* Design for Testability
* Anforderungen für Application Monitoring
* Architekturrelevante Anforderungen
* Anforderungen an die Hardware
* Anforderungen an das Netzwerk
* Anforderungen an den Betrieb

## Inhalte Umsetzungskonzept

Spezielle Lösungsthemen der ITSV GmbH:

* Mengengerüst
* Grundlegende architekturelle Überlegungen
* Identifikation und Bewertung möglicher unterschiedlicher Lösungswege
* Definition der Abnahmekriterien
* Umgang mit Standard Request Headern
* Nutzung standardisierter Lösungen/Komponenten
* Erstellung standardisierter Lösungen/Komponenten
* Austauschbarkeit
* Erforderliche Maßnahmen für Business Continuity
* Definition der Schnittstellen
* Einhaltung des Schutzbedürfnisses des Produkts (Informationssicherheit)
* Umfang und Art der erforderlichen Benutzerdokumentation
* Vorgaben zur Testautomatisierung (Grad der Testautomatisierung)
* Spezielle Anforderungen für die Betreibbarkeit und Verrechenbarkeit
* Spezielle Anforderungen an die ITSV-RZ-Autodeploy-Fähigkeit
* Spezielle Anforderungen für Application Monitoring
* Spezielle Anforderungen an Schnittstellen zu Webservices
* Spezielle Anforderungen an die Anbindung der Alarmierung
* Anforderungen an die Bereitstellung von Personal für Unterstützungsleistungen für den Kunden

Ein zweiter Teil des Umsetzungskonzepts ist die vertragliche und kommerzielle Sicht:

* Geschätzte Aufwände
* Erforderliche Unterstützungsleistung des Kunden
* Geschätzter Preis
* Folgekosten
* Grober Zeitplan
* Abnahmemodalitäten (siehe auch High Level Abnahmekriterien)

## Inhalte Design Review des Software Designs

Beispiel für Themen, die im Design Review überprüft werden sollen:

* Ist das Software Design so dokumentiert, das langfristig (mit Personen, die nicht in die Entwicklung involviert waren) eine langfristige Wartbarkeit und Erweiterbarkeit mit vertretbarem Aufwand möglich ist?
* Sind die ausgewählten Algorithmen geeignet, die Anforderungen an Performance zu erfüllen?
* Ist das System im Sinne der prognostizierten Mengengerüste einfach skalierbar?
* Ist die System- und Softwarearchitektur geeignet, die Zuverlässigkeitsanforderungen zu erfüllen?
* Sind das System und die verwendeten Protokolle, Umgebungen und Algorithmen geeignet, die Sicherheitsanforderungen zu erfüllen?
* Sind die Anforderungen an das Design for Testability eingehalten?
* Sind die Anforderungen an das Design für den späteren Betrieb eingehalten?

# Anhang B: Arten der Überprüfung

## Existenzprüfung (Sichtkontrolle)

Sichtkontrolle oder automatisierte Kontrolle, ob das angesprochene Objekt vorhanden ist.

|  |  |
| --- | --- |
| Ergebnis | ja |
|  | nein |

## Formales Review

Manuelle Überprüfung, ob die formalen Anforderungen an das angesprochene Objekt erfüllt sind.

|  |  |
| --- | --- |
| Ergebnis | erfüllt |
|  | unbedeutende Abweichungen ohne Korrekturerfordernis |
|  | Abweichungen mit Korrekturerfordernis ohne Wiederholungsreview |
|  | Abweichungen mit Korrekturerfordernis mit Wiederholungsreview |

## Inhaltliches Review

Manuelle Überprüfung durch einen oder mehrere Prüfer, welche die Einhaltung der definierten Review-Kriterien für das angesprochene Objekt kontrollieren und bewerten. Ob die Kriterien erfüllt sind, liegt weitgehend im Ermessensspielraum der Prüfer und erfordert von ihnen eine verantwortliche Bewertung des aufgrund von Abweichungen bestehenden Restrisiken. Die Prüfer müssen ausreichend qualifiziert sein, um den Inhalt des Objekts bewerten zu können.

|  |  |
| --- | --- |
| Ergebnis | erfüllt |
|  | unbedeutende Abweichungen ohne Korrekturerfordernis |
|  | Abweichungen mit Korrekturerfordernis ohne Wiederholungsreview |
|  | Abweichungen mit Korrekturerfordernis mit Wiederholungsreview |

## Integrationstest

Der Integrationstest prüft dynamisch, ob die Interaktion zwischen zwei oder mehreren Software-Komponenten korrekt funktioniert. Basierend auf den definierten Testfällen kann der Integrationstest manuell oder automatisiert durchgeführt werden.

|  |  |
| --- | --- |
| Ergebnis | Testendekriterien erfüllt |
|  | Testendekriterien nicht erfüllt |

## Überprüfung der Nachweise

Bei der Überprüfung von Nachweisen werden Existenz und Plausibilität (Korrektheit) der Nachweise sowie Aktualität und Vollständigkeit überprüft.

|  |  |
| --- | --- |
| Ergebnis | vorhanden und korrekt |
|  | unbedeutende Abweichungen ohne Korrekturerfordernis |
|  | Abweichungen mit Korrekturerfordernis ohne erneute Überprüfung |
|  | Abweichungen mit Korrekturerfordernis mit erneuter Überprüfung |

## Test

Der Test führt definierte Testfälle aus und prüft, ob die Testendekriterien erfüllt sind.

|  |  |
| --- | --- |
| Ergebnis | Testendekriterien erfüllt |
|  | Testendekriterien nicht erfüllt |

## Lessons Learned

Bei Lessons Learned werden die Erfahrung aus dem Vorhaben besprochen und bewertet. Als Ergebnis werden Änderungsmaßnahmen definiert.

|  |  |
| --- | --- |
| Ergebnis | nicht durchgeführt  durchgeführt  durchgeführt und dokumentiert |

## Vergleich

Zwei Ergebnisse werden miteinander v(qualitativ) erglichen.

|  |  |
| --- | --- |
| Ergebnis | gleich |
|  | unbedeutende Abweichungen („qualitativ gleich“) |
|  | Bedeutende Abweichungen („ungleich“) |

## Ausbildungsnachweis

Ein formaler Nachweis der Qualifikation einer Person ist vorhanden (oder nicht).

|  |  |
| --- | --- |
| Ergebnis | vorhanden |
|  | Nicht vorhanden |

# Anhang C: Q-Kriterien für Artefakte/Aktivitäten

## Aktualität

Es wurde nachweislich überprüft, dass das Artefakt aktuell und auf dem gleichen Stand wie verwandte/referenzierte Artefakte ist.

## Berücksichtigung im Design

Es existiert ein Nachweis, dass die Vorgaben beim Design der Software berücksichtigt wurden.

## Eignung und Umsetzbarkeit

Die Vorgaben sind geeignet, das zugehörige Ziel zu erreichen und können mit wirtschaftlich vertretbarem Aufwand umgesetzt werden.

## Existenz

Es existiert ein Nachweis der Existenz des angeführten Artefakts/der angeführten Aktivität.

## Existenz der Entscheidung

Es existiert eine nachweisliche Entscheidung, dass ein Aspekt im Artefakt/in der Aktivität berücksichtigt werden muss.

## Existenz Mindestinhalt

Es existiert ein Nachweis, dass das angeführte Artefakt/die angeführte Aktivität den definierten Mindestinhalt umfassen.

## Grad der Testabdeckung

Es existiert eine Dokumentation, in der der Grad der Testabdeckung in Bezug auf die alle Anforderungen und Erwartungen an das getestete Artefakt beschrieben ist. Dies kann in Prozent oder in definierten Stufen erfolgen.

## Inhaltliche Korrektheit

Es existiert ein Nachweis, dass das angeführte Artefakt/die angeführte Aktivität inhaltlich in Bezug auf die Anforderungen und Erwartungen der Stakeholder korrekt ist.

## Korrekte Funktionalität

Es existiert ein Nachweis, dass das angeführte Artefakt die korrekte Funktionalität besitzt.

## Nachweis der Einhaltung

Es existiert ein Nachweis, dass die Vorgaben eingehalten werden.

## Nachweis der Nutzung

Es existiert ein Nachweis, dass die zugrundeliegenden Artefakte für die Entscheidung verwendet wurden.

## Qualifikation des durchführenden Personals

Es existiert ein Nachweis, dass das mit der Aufgabe betrautet Personal für die Durchführung ausreichend qualifiziert ist.

## Qualitative Übereinstimmung

Es existiert ein Nachweis, dass die verglichenen Artefakte qualitativ übereinstimmen, d. h. kein grundlegend unterschiedliches Verhalten oder unterschiedliche Größenordnungen aufweisen.

## Risikobasierte Berücksichtigung

Es existiert ein Nachweis, dass die definierten Aspekte das angeführte Artefakt/der angeführten Aktivität in den wesentlichen Risikoaspekten charakterisieren und dass diese Aspekte im Artefakt/in der Aktivität berücksichtigt sind.

## Übereinstimmung mit Anforderungen und Erwartungen

Es existiert ein Nachweis, dass das Artefakt/die Aktivität die impliziten und explizit definierten Anforderungen und Erwartungen der Stakeholder erfüllt.

## Vollständige Berücksichtigung

Es existiert ein Nachweis, dass die definierten Aspekte das angeführte Artefakt/der angeführten Aktivität vollständig charakterisieren und dass alle diese Aspekte im Artefakt/Ablauf berücksichtigt sind.

## Vollständigkeit

Es existiert ein Nachweis, dass das angeführte Artefakt/die angeführte Aktivität in Bezug auf die Anforderungen und Erwartungen der Stakeholder vollständig ist.

# Anhang D: Referenzierte Rollen

* Administrationspersonal
* Applikationsmanagement
* Auditor
* Betrieb
* Betrieb des Kunden
* Controlling
* Endbenutzer
* Entwicklung
* Experten
* Integrationsteam
* Interner Auftraggeber (Management)
* IT-Architektur
* Kunde
* Lösungsdesign
* Multiprojektmanagement
* Personal des Service Desks
* Product Owner
* Produktmanagement
* Projektteam
* Prozessverantwortlicher
* Reviewer
* Service Management
* Stabsstelle Informationssicherheit
* Stabsstelle Qualität
* Tester
* Testteam
* Umsetzungsverantwortlicher
* Verrechnung
* Zentrale Release-Koordination

1. In der ITSV GmbH existieren derzeit keine einheitlichen Vorgaben für Zielwerte. Bis auf wenige Aus­nah­men sind alle Kriterien informell und qualitativ. Die hier angeführten Werte sind also Vorschläge, die sei­tens der ITSV GmbH in der Folge adaptiert und verbindlich festgelegt werden müssen. [↑](#footnote-ref-1)
2. Es wird dringend empfohlen, derartige WIKI-Seiten einer regelmäßigen redaktionellen Überprüfung zu unterwerfen. [↑](#footnote-ref-2)