

Dokumentation ITIL - Qualitätssicherung

Lara Krautmacher
lara.krautmacher@student.reutlingen-
university.de
Matrikelnummer: 123425
Informatik: IT-Management
72762 Reutlingen
Baden-Württemberg, Deutschland

René Wikow
rene.wiskow@student.reutlingen-
university.de
Matrikelnummer: 801861
Informatik: IT-Management
72762 Reutlingen
Baden-Württemberg, Deutschland

Elena Kirsch
elena.kirsch@student.reutlingen-
university.de
Matrikelnummer: 763207
Informatik: IT-Management
72762 Reutlingen
Baden-Württemberg, Deutschland

ABSTRACT

Qualitätssicherung spielt eine zentrale Rolle in jedem Unternehmen. Reibungslose Abläufe interner Prozesse sowie die Erfüllung von Ansprüchen an ein Endprodukt werden durch die Qualitätssicherung garantiert. Im Zuge dieser Seminararbeit wird anhand eines Planspiels evaluiert inwieweit die IT Infrastructure Library die Organisation und Durchführung einer effizienten Qualitätssicherung unterstützt. Dazu wurde das fiktive Unternehmen *TopBlogAG*, bestehend aus fünf Arbeitsgruppen, gegründet, welches als Zielsetzung die Erstellung einer Blogging-Plattform hat. Die Arbeitsgruppe Qualitätssicherung hat dazu verschiedene Teststrategien sowie Schwachstellenanalysen durchgeführt. Durch die Definition der Key Performance Indicators in der Service-Quality Policy ist festgelegt, wann die Ziele des Qualitätssicherungsprozesses erreicht sind.

KEYWORDS

IT-Management, ITIL, Qualitätssicherung, Service Transition

1 EINLEITUNG - ELENA

Unternehmen sind bestrebt ihre Prozesse, Produkte oder auch Dienstleistungen durch Frameworks und Zertifizierungen zu verbessern und deren Qualität stetig zu steigern. Die Qualität von Produkten ist entscheidend wie gut sich ein Unternehmen in dem jeweiligen Markt etablieren kann. Um im Bereich des IT Service Management (ITSM) einen messbaren Qualitätsstandard zu erhalten, wurde im Jahr 2005 die ISO/IEC 20000 IT Service veröffentlicht¹. Durch ITSM stellen Organisationen sicher, dass ihre IT-Services so funktionieren, dass diese durch Benutzer*innen und das Unternehmen selbst genutzt werden können². Im Allgemeinen handelt es sich bei ITSM um eine Reihe von Richtlinien und Praktiken, die die Implementierung, Bereitstellung und Verwaltung von IT-Services festschreibt. Dabei liegt der Fokus auf den erklärten Bedürfnissen der Endnutzer*innen und den erklärten Zielen des Unternehmens. Für die Implementierung und Dokumentation von ITSM dient das am weitesten verbreitete Best-Practice-Framework, die Information Technology Infrastructure Library (ITIL)³. Das Framework ITIL bietet den Unternehmen die Rahmenbedingungen, um die Services optimal auf die Anforderung aus dem Business abzustimmen und regelmäßig auf die beste Unterstützung der Geschäftsprozesse zu überprüfen [1]. ITIL wird bei seiner Anwendung in

fünf Lebenszyklusphasen eingeteilt. Die verschiedenen Lebenszyklusphasen sind gegliedert nach welchen Objekten sich die jeweilige Phase gerade richtet, welche Schlüsselkonzepte verwendet, welche Prozesse angewendet werden und welche Modelle in dem Zyklus zugrunde gelegt werden. Zudem sind die jeweiligen Dokumente und Ergebnisse jeder Phase festgelegt. Innerhalb der Service Transition sowie der Continual Service Improvement spielt die Qualitätssicherung eine wichtige Rolle. Somit müssen die in der Service Strategy identifizierten Services und Strategien in der Service Transition Phase getestet werden. Und auch die Ergebnisse aus der Service Design Phase wie zum Beispiel die Architekturen oder die Service-Level-Agreements müssen durch die Qualitätssicherung überprüft werden[1].

2 THEORETISCHE HINTERGÜNDE - ELENA

ITIL ist toll

2.1 ITIL

ITIL ist toll

2.2 Qualitätssicherung in ITIL - Lara

Qualitätssicherung spielt im ITIL Framework eine zentrale Rolle. Dies ist darauf zurückzuführen, dass ein Grundgedanke von ITSM darin besteht die Qualität und Quantität von IT-Services zu planen, überwachen und steuern [1]. An dieser Stelle wird beschrieben in welchen Bereichen von ITIL Qualitätssicherung beachtet werden muss. ITIL Service Management ist in fünf Kernbereiche aufgeteilt, Service Strategy, Service Design, Service Transition, Service Operation, welche zeitlich nacheinander ablaufen, und Continual Service Improvement (CSI), welcher parallel zu den anderen Teilbereichen kontinuierlich bespielt wird. Qualitätssicherung findet dabei in den größten Teilen im Bereich CSI statt, muss jedoch auch in den anderen Teilbereichen immer mit betrachtet werden. Bei *Service Strategy*, *Service Design*, *Service Transition*, *Service Operation* wird Qualitätsmanagement in die Prozesse integriert, indem bei der Planung zunächst betrachtet wird, welche Qualitätsanforderungen an den Service bestehen, sodass dem Nutzer einen Mehrwert im Produkt erkennt. Diese Anforderungen werden anschließend in messbaren Kennzahlen (KPI's) quantifiziert. Die Definition der KPI's, die gemessen werden sollen bilden die Grundlage für den CSI-Improvement Prozess. In diesem werden zur Verbesserung des Services sieben Phasen durchlaufen.

2.2.1 Qualitätssicherung Allgemein. bdbjsj

2.2.2 Continuous Service Improvement.

¹<https://www.iso.org/standard/51986.html>, Zugriff: 31.05.2021

²<https://www.ibm.com/cloud/learn/it-service-management-toc-what-is-it-CCWD9gs4>, Zugriff: 31.05.2021

³<https://www.ibm.com/cloud/learn/it-service-management-toc-what-is-it-CCWD9gs4>

2.2.3 Testing. Der Testing Prozess stellt die Einhaltung des Vertrags, welcher mit dem Kunden über die Qualitätsanforderungen abgeschlossen wird, sicher. Der Nutzen des Kunden steht daher im Mittelpunkt. Es kann dabei zwischen zwei verschiedenen Kategorien unterschieden werden: Die *Utility* (Nützlichkeit) und die *Warranty* (Garantie) [1]. Die *Utility* befasst sich dabei direkt mit den vom Kunden erwarteten Services, während die *Warranty* die Verfügbarkeit dieser Services ins Auge fasst. Aus diesen beiden Kategorien leitet sich die Notwendigkeit ab, zum einen das Produkt selbst zu Testen, als auch die Strukturen, welche den Nutzen des Produkts ermöglichen. Dazu gehören unter anderem interne Services, sowie das Deployment- Umfeld. Die Tests zielen dabei darauf ab, Fehler zu identifizieren und zu evaluieren ob vom Kunden gestellte Anforderungen wie gewünscht erfüllt wurden. Final müssen die erhobenen Ergebnisse an die verantwortliche Personen gemeldet werden, um Fehlerquellen möglichst schnell zu beheben. Da in vielen Bereichen getestet werden muss, um diese Kriterien sicherzustellen, ist eine umfassende Teststrategie unabdingbar.

Bei der Entwicklung einer geeigneten Teststrategie wurde sich an das Service V-Modell gehalten, welches in Abbildung 1 dargestellt ist. Dieses definiert zu jedem Design- bzw. Entwicklungsschritt der einzelnen Services eine passenden Test. Dabei kann dieser Prozess in fünf Abschnitte unterteilt werden:

Der *Component Test* wird parallel zur eigentlichen Service Entwicklung definiert. Dieser prüft, ob die einzelnen Komponenten, bzw. eine Gruppe an Komponenten, die geforderte Spezifikationen erfüllt. Es wird die direkte technische Umsetzung der Services getestet. Der *Release Package Test* testet anschließend, ob die erstellten Services fehlerfrei bereitgestellt (deployed) werden können. Der *Operational Readiness Test* bezieht sich auf die Ressourcen, welche vorhanden sein müssen, um den Service bereitzustellen. Damit sind nicht nur technische, sondern auch Humanressourcen gemeint. Es wird also genau so danach gefragt, ob genügend Speicher vorhanden ist, wie auch, ob der Support für den jeweiligen Service geleistet werden kann. Der *Acceptance Test* erfolgt im direkten Zusammenspiel mit dem Kunden. Hier wird sichergestellt, dass der Service alle gestellten Kundenanforderungen zufriedenstellend erfüllt. An letzter Stelle steht die *Service Validierung*. Dazu wird geprüft, ob der Service alle Verträge einhält.

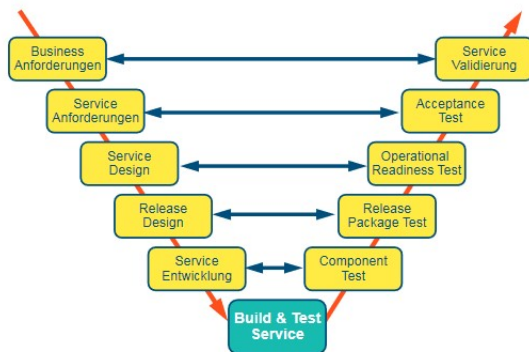


Figure 1: Service V-Modell [1]

3 PRAKTISCHE DURCHFÜHRUNG

Macht Spaß

3.1 Qualitätssicherung im Planspiel - Lara

Macht Spaß

3.1.1 Service Quality Policy. Definition Key Performance Indicators und Testing der KPI's.

3.1.2 Prozessdefinition zur Qualitätssicherung. Hier kommt ein Text

3.2 Testing-Rene

In diesem Abschnitt wird aufgeführt, wie mithilfe des Service V-Modells Tests entwickelt und durchgeführt wurden.

3.2.1 Component Test. Die Component Tests beziehen sich wie in Abschnitt 2.2.3 beschrieben auf die direkte Implementierung der Service-Anforderungen. Die Frage, welche die Component Tests beantworten sollen lautet daher: Werden die Anforderungen in der Implementation zufriedenstellend umgesetzt? Um diese Frage zu beantworten zunächst für jede Anforderung an den Service eine Schwachstellenanalyse durchgeführt. Diese Schwachstellen wurden anschließend mit einem Risikostatus gewichtet. Dabei wurden vier verschiedene Gewichtungen definiert, welche hier aufsteigend aufgelistet sind: *Geringfügig*, *Mittelmäßig*, *Kritisch* und *Katastrophal*. Diese Gewichtung wurde vorgenommen, um die Dringlichkeit einer Anpassung zu dokumentieren. Zu jeder Anforderung wurde weiterhin ein oder mehrere Annahmekriterien definiert, welches sich direkt aus dem vom Business Team erstellten Service Portfolio ableiten. Im folgenden wurden die Component Tests durch den Softwaretester durchgeführt und dokumentiert, ob der Test bestanden wurde und von wem er durchgeführt wurde. Weiterhin wurden Auffälligkeiten genau dokumentiert, um eventuelle Fragen zu klären oder mehr Informationen zu gescheiterten Tests zu liefern. Die Testergebnisse wurden gebündelt für einzelne Komponenten an den *Head of Development* zurückgemeldet. Weiterhin wurde eine Metrik für die Qualität einzelner Komponenten eingeführt. Nach dieser hat eine Komponente eine Qualität von 100% , wenn alle auf ihr ausgeführten Tests bestanden sind. Bei der Berechnung der Metrik wird die Gewichtung des Risikos mit einbezogen indem den einzelnen Gewichtungen aufsteigend Werte von eins bis vier zugewiesen wurden. Wurde ein Test nicht bestanden, wird der gewichtete Anteil von den 100% abgezogen. Als zufriedenstellende Qualität einer Komponente wurde ein Wert von über 90% festgelegt. Eine Beispieldokumentation für einen Component Test findet sich in Tabelle 1.

- Weiterhin wurde der interne Service des Service Desks, das Redmine Portal, getestet. Hier wurde ähnlich vorgegangen

3.2.2 Release Package Test.

- Release- und Deployment Manager verantwortlich?
- Funktionieren Release Optionen? Funktioniert Deployment?

3.2.3 Operational Readiness Test.

- Sind wir in der Lage den Service so zu betreiben wie der Kunde es erwartet? Warranty

Komponente	Annahmekriterium	Schwachstellenanalyse	Risikobewertung	Testergebnis	Verantwortlich	Anmerkungen
Registrierung	Reg. über: E-Mail Passwort Name Nachname Benutzername	Es können Texte welche keine E-Mail Adresse sind als E-Mail angegeben werden.	Katastrophal	Bestanden	Software-Testerin Lara Krautmacher	-
Registrierung	Person bekommt nach Registrierung einen Bestätigungslink per Mail	Link in E-Mail ist nicht gültig oder E-Mail kommt nicht an.	Kritisch	Bestanden	Software-Testerin Elena Kirsch	-
...

Table 1: Component Tests Registrierung

- Fähigkeiten?
- Ressourcen? (iT Operations)
- Mitarbeiter?
- Support organisiert? (Service Desk)

3.2.4 Acceptance Test.

- Erfolgt mit Kunden
- Service Abnahme wird mit Kunden durchgegangen (Business Team)

3.2.5 Service Validierung.

- Validierung der Angebote und Verträge
- Werden Verträge eingehalten?

3.3 Testplanung und -durchführung - Elena

Excel

4 REFLEKTION DES PROJEKTS

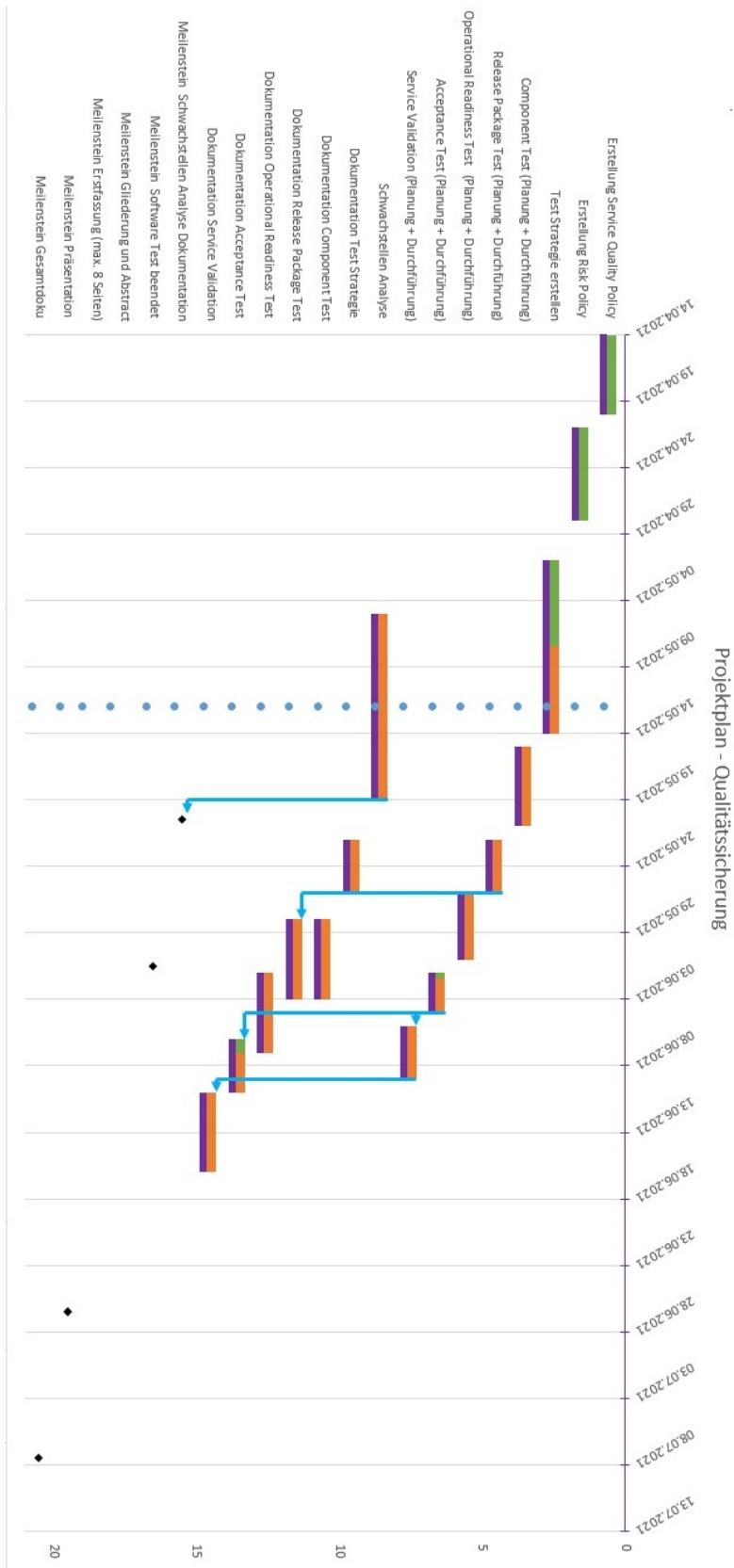
Testen ...

5 FAZIT - RENE

A ANHÄNGE

REFERENCES

- [1] Martin Beims and Michael Ziegenbein. 2015. *IT-Service-Management in der Praxis mit ITIL: Der Einsatz von ITIL Edition 2011, ISO/IEC 20000:2011, COBIT 5 und PRINCE2* (4., überarbeitete und erweiterte auflage ed.). Hanser, München.



4
Figure 2: Projektplan - Qualitätssicherung