

Institut für Informatik  
Lehrstuhl für Programmierung und Softwaretechnik

LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN



**Seminararbeit**

# **Neue Fachmeinungen zum Thema Dokumentation**

**Christian Ungnadner**

Informatik Master

Aufgabensteller: Dr. Frank Sarre  
Abgabetermin: 29. Januar 2017



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Problemstellung . . . . .	1
1.3	Ziel der Arbeit . . . . .	1
<b>2</b>	<b>Dokumentation</b>	<b>3</b>
2.1	Definition . . . . .	3
2.1.1	Gesetze . . . . .	3
2.1.2	Normen . . . . .	3
2.1.3	Fachliteratur . . . . .	4
2.1.4	Zusammenfassung . . . . .	4
2.2	Unterschiedliche Arten von IT-Dokumentation . . . . .	4
2.2.1	Projektmanagementdokumentation . . . . .	4
2.2.2	Anwenderdokumentation . . . . .	4
2.2.3	Programm- /Entwicklungsdokumentation . . . . .	5
2.2.4	Quellcodedokumentation . . . . .	5
2.2.5	Testdokumentation . . . . .	5
2.2.6	Weitere Dokumentationsarten . . . . .	6
2.2.7	Zusammenfassung . . . . .	6
<b>3</b>	<b>Neue Fachmeinungen zum Thema Dokumentation</b>	<b>7</b>
3.1	Dokumentation in Softwareerstellungsverträgen . . . . .	7
3.2	Projektmanagement und die zugehörige Dokumentation . . . . .	9
3.3	Software-Anforderungsdokumentation . . . . .	11
<b>4</b>	<b>Zusammenfassung</b>	<b>15</b>
	<b>Abbildungsverzeichnis</b>	<b>17</b>
	<b>Literaturverzeichnis</b>	<b>19</b>



# Kapitel 1

## Einleitung

Im Rahmen dieser Arbeit wird auf das Thema Dokumentation in IT-Projekten eingegangen. Angefangen mit einer Motivation und der Problemstellung sowie der Zielsetzung. Im Hauptteil folgt dann eine allgemeine Einführung in die verschiedenen Dokumentationsarten um einen grundlegenden Überblick zu verschaffen. Im Anschluss daran werden drei Fachartikel zusammengefasst und deren Neuerungen dargestellt. Abgerundet wird das Ganze durch eine Zusammenfassung.

### 1.1 Motivation

Aufgrund des anhaltenden Anstiegs an Komplexität und Umfang von Softwareprodukten spielt die Dokumentation eine immer wichtiger werdende Rolle, sowohl während der Projektumsetzung, als auch nach Abnahme der fertigen Software bzw. des fertigen Systems.

### 1.2 Problemstellung

Nicht selten kommt es während IT-Projekten zu Zeitproblemen oder auch Budgetengpässen, dies führt häufig dazu, dass der Dokumentation vom Projektverantwortlichen eine minderwertigere Rolle zugewiesen wird oder in Extremfällen komplett weggelassen wird.

### 1.3 Ziel der Arbeit

Im Rahmen dieser Arbeit wird das Thema Dokumentation innerhalb von IT-Projekten genauer betrachtet. Es wird erläutert welche Arten existieren, welche Probleme bestehen, welche Rolle die gegebene Rechtsprechung einnimmt und der aktuelle Stand der Dinge ist. Daraufhin werden drei Fachartikel, welche sich mit einzelnen Bereichen der Dokumentation befassen, zusammengefasst und mögliche Erkenntnisse, Neuerungen oder Verbesserungsvorschläge der Autoren dargestellt.



# Kapitel 2

## Dokumentation

Der Begriff Dokumentation ist *eigentlich* klar definiert. Der Duden beschreibt das dokumentieren z.B. als „Zusammenstellung und Nutzbarmachung von Dokumenten, Belegen und Materialien jeder Art“ [dok].

### 2.1 Definition

Um für den Begriff Dokumentation im IT-Bereich eine Definition zu finden wird es schon schwierig, denn es gibt *eigentlich* keine, was dazu führt, dass es hier keine einheitliche Sprache gibt. In der Praxis werden deshalb verschiedene Quellen bzw. Dokumenttypen zu Hilfe genommen, um ein halbwegs einheitliches Verständnis zu vermitteln. Anbei wird auf die, für diese Arbeit relevantesten, eingegangen.

#### 2.1.1 Gesetze

Wie man Reis et al. [RR09] entnehmen kann, ist es schwer für den IT-Bereich Gesetze oder Verordnungen zu finden, denn es gibt sie nicht. Es gibt jedoch abgeleitete Gesetze, welche bei Gerichtsurteilen bzgl. IT-Dokumentation Verwendung fanden. Das wären beispielsweise das Produktsicherheitsgesetz (ProdSG). Dieses verschafft Klarheit über sicherheitsrelevante Gesichtspunkte bei der Markteinführung neuer Produkte (u.a. Software, Hardware, o.ä.). Außerdem gibt es noch das Produkthaftungsgesetz (ProdHaftG), z.B. ist eine technische Dokumentation/Betriebsanleitung fehlerhaft und führt zu einer Art von Schaden, kann der Hersteller zur Haftung gezogen werden [teca, tech].

#### 2.1.2 Normen

Normen haben sich schon seit längerem in IT-Unternehmen als eine Art Standard etabliert. Eine der bekanntesten Normreihen ist die ISO-9000 [90096], diese definieren Standards und Regeln vor allem für den Bereich des sog. Qualitätsmanagements und ist bei etlichen Auftraggebern als eine Voraussetzung für einen potentiellen Auftragnehmer anzusehen, d.h. der Auftragnehmer muss ISO-9000 zertifiziert sein [RR09]. Hier gilt es allerdings zu beachten, dass es verschiedene Organisationen gibt, welche sich mit Normen befassen, z.B. ISO, DIN, IEC, IEEE, etc. DIN ist z.B. ein deutscher Standard, wohingegen ISO, das internationale Gegenstück ist. Das bedeutet aber nicht automatisch, dass es zu jeder ISO-Norm auch eine DIN-Norm oder umgekehrt gibt. Ähnlich ist das mit den anderen Organisationen, dies hat zur Folge, dass etliche Normen nur in englischer oder französischer Sprache existieren. Außerdem hat sich in diesem Bereich

in den letzten Jahren viel getan, da die Wichtigkeit von Dokumentation jeglicher Art zu nimmt, wodurch sich folgern lässt, dass es mehr Organisationen gibt, welche Standards in bestimmten Bereichen etablieren wollen. Dies trägt wiederum dazu bei, dass das „Angebot“ an Normen unübersichtlicher wird und die Etablierung deutlich erschwert [RR09].

### 2.1.3 Fachliteratur

Ein weiterer Bereich ist die Fachliteratur. Diese umfasst Publikationen, welche mittlerweile als eine Art Standard, bezogen auf Kennzeichen von IT-Systemen, gesehen werden. Die eben angesprochenen Publikationen dienen ebenso als Quelle für die zuvor erwähnten Normen. Ein bekanntes Buch im Dokumentationsumfeld, ist beispielsweise das „Praxisbuch IT-Dokumentation“ [RR09].

### 2.1.4 Zusammenfassung

Insgesamt lässt sich folgern, dass es etliche unterschiedliche Quellen gibt, in welchen unterschiedliche Arten von Dokumentationen im IT-Bereich niedergeschrieben sind.

## 2.2 Unterschiedliche Arten von IT-Dokumentation

Im folgenden Abschnitt wird auf einzelne Arten der Dokumentation eingegangen. Es wird erläutert welchen Umfang diese beinhalten und welche Probleme diese ggf. mitbringen. Jedoch wird hier nicht auf jede existierende Art von Dokumentation im IT-Umfeld eingegangen, da der Rahmen dieser Arbeit das nicht hergibt.

### 2.2.1 Projektmanagementdokumentation

Die Projektmanagementdokumentation oder im Volksmund auch Projektdokumentation genannt, ist eine der wichtigsten Dokumentationsarten, z.B. um nach dem Scheitern eines Projekts rekonstruieren zu können, wieso dieses gescheitert ist und wer verantwortlich dafür ist. Außerdem kann eine Projektdokumentation auch im Falle eines Rechtsstreits von Vorteil sein [RR09]. Die DIN-Norm 69901 [Deu09b] befasst sich beispielsweise mit der Projektdokumentation (nicht zu Verwechseln mit Projektmanagementhandbuch, Projekthandbuch oder Projektplan) und bezeichnet sich selbst als „*Zusammenstellung ausgewählter, wesentlicher Daten über Konfiguration, Organisation, Mitteleinsatz, Lösungswege, Ablauf und erreichte Ziele des Projektes*“ [Deu09b]. Wichtig ist, dass dies keine Verbindlichkeit ist. Eine andere Möglichkeit ist, dass die Dokumentationsart von dem verwendeten Projektvorgehensmodell festgelegt bzw. beschrieben wird. Eine weitere Möglichkeit, welche auch häufiger vorkommt ist, dass IT-Abteilungen eigene Projektmanagementhandbücher erstellen, in welchen festgelegt ist, welche Dokumentationen zu erstellen sind bzw. ganz allgemein welche Normen oder Richtlinien innerhalb eines Projektes eingehalten werden müssen.

### 2.2.2 Anwenderdokumentation

Alternativen für den Begriff Anwenderdokumentation sind beispielsweise Benutzerhandbuch, Anwenderhandbuch, Programmhandbuch, Bedienungsanleitung oder auch Gebrauchsanweisung. Diese Begriffe finden in verschiedenen Fällen Verwendung wie z.B. Gesetzestexten, Normen o.ä. Die Rechtssprechung besagt an dieser Stelle, dass zu

jedem Softwareprodukt auch ein Benutzerhandbuch vom Softwarelieferanten zu erbringen ist. Aus rechtlicher Sicht ist der Begriff Bedienungsanleitung ein Synonym, somit gilt folgender Gesetzestext: „*Eine Kaufsache ist mangelhaft im Sinne des § 434 BGB, wenn die Bedienungsanleitung in wesentlichen Punkten unvollständig oder fehlerhaft ist, sodass bei entsprechendem Gebrauch der - ansonsten einwandfreien - Kaufsache, Fehlfunktionen auftreten.*“ vgl. OLG München, Urteil vom 09.03.2006, Az. 6 U 4082/05 [bed]. Um die Frage nach einem Inhalt beantworten zu können, dient z.B. ISO/IEC/IE-EE 15289 [61011]. Folgende Punkte soll laut dieser Norm eine Anwenderdokumentation beinhalten:

- Kurzbeschreibung der Zweckbestimmung des Systems
- Hinweise auf Risiken, Warnhinweise, Sicherheitsmaßnahmen
- Informationen zur Betriebsumgebung
- Bedingungen um die spezifizierten Aufgaben am System ausführen zu können
- Problem- und Fehlerbehebungsverfahren
- Support
- Verfahren um auf das System zugreifen und es wieder verlassen zu können
- Auflistung und Erläuterungen von Systemmeldungen

Eine weitere Norm ist ISO/IEC 26514 [57111], diese beschreibt ebenso detaillierte Auflagen zum Inhalt. Außerdem geht hier auch hervor, dass es unterschiedliche Benutzergruppen gibt, für welche ggf. verschiedene Dokumentationen anzufertigen sind [57111].

### 2.2.3 Programm- /Entwicklungsdokumentation

Bis 2004 existierte DIN-66230 [din81], welche den Inhalt ziemlich genau beschrieben hat. Jedoch wurde sie zurückgezogen, da sie für nicht mehr zeitgemäß gehalten wurde. DIN-66230 besagte, dass eine Programmdokumentation „alle“ Informationen zu einer Software enthalten muss. Mittlerweile finden sich diese Informationen aber in anderen Abschnitten wieder [Sof, din81]. Heute wird als Programmdokumentation meistens UML verwendet, welche unter anderem auch in der Norm ISO/IEC 19501 [Sta05] hinterlegt ist. Die Modellierungssprache hilft bei der Spezifikation, Konstruktion sowie der Dokumentation von Software [Sta05, uml].

### 2.2.4 Quellcodedokumentation

Ob eine Quellcodedokumentation geschuldet ist oder nicht, hängt erst einmal davon ab, ob der Sourcecode überhaupt an den Auftraggeber (ggf. auch Dritte) weitergegeben werden muss. Ist dies der Fall, ist es laut einem Gerichtsurteil des Amtsgerichts Pforzheim (Urteil vom 7. Juli 1987, 3 C 540/86) [sys]) äußerst wahrscheinlich, dass diese Dokumentation auch geschuldet ist. Was genau diese Art von Dokumentation beinhalten soll, ist nicht genauer definiert.

### 2.2.5 Testdokumentation

Der Standard IEEE 829-2008 [IEE08] Standard for Software Test Documentation beschreibt acht Dokumente, welche eine Testdokumentation enthalten muss.

1. Testplanung
2. Testdesign
3. Testspezifikation
4. Testfallspezifikation
5. Testablaufspezifikation
6. Testprotokolle
7. Testfehlerbericht
8. Testabschlussbericht

Ob eine Testdokumentation jedoch geschuldet ist, hängt von den Vertragsvereinbarungen ab [IEE08].

### 2.2.6 Weitere Dokumentationsarten

Da im Rahmen dieser Arbeit nicht alle Dokumentationsarten beschrieben werden können, wurden nur die „wichtigeren“ Arten näher erläutert. Anbei folgt noch eine Auflistung weiterer Dokumentationen. Aufgrund ihrer Namen, kann ihr Inhalt meist abgeleitet werden: *Online-Hilfe*, *Installationsanleitung*, *Betriebsanleitung*, *Systemdokumentation*, *Schulungsdokumentation* .

### 2.2.7 Zusammenfassung

Die zuvor angesprochenen Abschnitte zeigen, wie es mit den einzelnen Arten der Dokumentation steht. Es gibt viele, alle sind mehr oder weniger dokumentiert. Teilweise existieren Normen bzw. Standards dazu, welche sich aber auch überschneiden können. Manche sind von Seiten der Rechtssprechung verpflichtend, selbst wenn vertraglich nichts dazu festgelegt wurde, andere nicht. Somit sollten sich Auftraggeber und Auftragnehmer vorab Gedanken machen, welche Arten von Dokumentation für sie notwendig sind und dies vertraglich festhalten, wenn möglich auf eine evtl. existierende Norm, welche die relevanten Kriterien erfüllt, zurückgreifen. Selbst eine vertragliche Regelung, kann aber noch zu Problemen führen, denn etliche Normen legen die Art des Inhalts fest, allerdings nicht den Detailgrad.

## Kapitel 3

# Neue Fachmeinungen zum Thema Dokumentation

In diesem Kapitel geht es um „*Neue Fachmeinungen zum Thema Dokumentation*“. In den vorherigen Abschnitten wurden verschiedene Dokumentationsarten sowie damit einhergehende Probleme angesprochen. Nun werden drei Artikel von Spezialisten (IT-Sachverständige) zusammengefasst und deren Erkenntnisse dargestellt.

### 3.1 Dokumentation in Softwareerstellungsverträgen

Als erstes dient der Artikel *Dokumentation Softwareerstellungsverträgen* von Sabine Schreiber-Ehle, Sachverständigerin für Systeme und Anwendungen, der Informationsverarbeitung insbesondere Systemsicherheit und Softwareentwicklung in der Fachzeitschrift *Computer und Recht* [SE15]. Frau Schreiber-Ehle geht in ihrem Artikel auf die Hintergründe sowie die Bedeutung dieser Dokumentation ein.

Im Anschluss an die Übergabe einer Software fällt dem Erwerber oft auf, dass ihm Dokumentation zur Software fehlt. Daraus ergibt sich laut der Autorin, dass vorab geklärt werden sollte, was die Absichten des Erwerbers sind. In Abhängigkeit des Verwendungszwecks der Software werden verschiedene Arten von Dokumentation notwendig, im Speziellen, wenn später Änderungen/ Anpassungen/ Neuerungen an der erhaltenen Software durch die eigene IT-Abteilung oder ggf. Dritte durchgeführt werden sollen. Als besonders wichtig wird erwähnt, dass die einzelnen Personengruppen, welche später bei einer Weiterverarbeitung beteiligt sind, identifiziert werden müssen. Denn diese benötigen auch unterschiedliche Arten an Dokumentation. Beispielsweise ist bei einer „Anpassbaren Software“ eine Anpassungsanleitung für die künftigen Entwickler von Bedeutung. Im Falle einer vollständigen Entwicklerübergabe, sind Dokumente wie Benutzerhandbuch, Anpassungsanleitung, Schnittstellendokumentation, Sourcecode-Dokumentation, etc. für Personen wie den zukünftigen Architekt, die Entwickler und das Betriebspersonal nötig. [SE15]

Des Weiteren geht die Autorin auch auf die Entwicklung der Softwareentwicklungsparadigmen ein, d.h. es existieren etliche verschiedene Dokumentationsarten nebeneinander. Für funktionsbezogene Programmierung existieren beispielsweise DIN 66001 [Her84] und DIN 66261 [din]. Für die objektorientierte Programmierung wurde die Dokumentationsnorm ISO/IEC 19505-2:2012 [19512] erstellt, die so genannte „*Unified Modelling Language*“ kurz UML [uml]. Mittlerweile sind Softwareprojekte aber so groß, dass die elementare Betrachtung einzelner Funktionen oder Programme nicht mehr ausreichend

ist. Es wird versucht das „Große Ganze“ zu betrachten. Hierfür existiert die Dokumentationsnorm ISO/IEC 19510:2013 *Business Process Modelling Language* [19513].

Im nächsten Abschnitt des Artikels erläutert Frau Schreiber-Ehle die zugrunde liegende Fachliteratur sowie Normen, auf welche in dieser Arbeit nicht detailliert eingegangen wird. Diese Dokumente beschreiben unzählige Arten mehr oder weniger genau zur Erstellung technischer Dokumentation [SE15].

Außerdem lobt sie so genannte „große Auftraggeber“, welche für die erfolgreiche Durchführung eigener Projekte zusätzliche Vorgaben an Dokumenten und Prozessen entwickelt haben. Des Weiteren wird von ihr das Vorgehensmodell *V-Modell XT* hervorgehoben, in Kapitel 5 des V-Modells wird nämlich beschrieben, wer in welcher Rolle, welche Dokumente mit was für einem Inhalt zu erstellen hat (geschuldete Dokumentation), außerdem wird der Umfang hierfür normalerweise im Liefervertrag festgelegt. [SE15]

Im Folgeschritt geht sie explizit auf ein Werk von *Clements et. al.* [GBI<sup>+</sup>10] ein, welches die vier elementaren Kategorien der technischen Dokumentation nennt:

1. Modulschichten: statischer Aufbau der Software
2. Komponenten- Beziehungsschicht: Beziehung zwischen einzelnen Teilen
3. Verteilungsschichten: Anordnung der Systemteile
4. sonstige Schichten: Interfacedokumentation, dyn. Systemverhalten

Zwei aus den ersten drei genannten Anforderungen, werden von *Clements et. al.* als minimal erachtet.

Im vierten Kapitel ihrer Ausarbeitung erläutert sie Praxisbeispiele, für unzureichende Dokumentation, auf welche hier nicht näher eingegangen wird [SE15].

Daraus resultiert eine Art Checkliste, welche Dokumentationen wann benötigt werden, um Probleme bzgl. mangelnder Dokumentation zu vermeiden. Hierfür relevante Punkte liefert die am Anfang erläuterte Textpassage, „was möchte der Auftraggeber nach Erhalt der Software machen und welche Personengruppen sind beteiligt“. Dafür bezieht sie erneut das Werk von *Clements et. al.* [GBI<sup>+</sup>10] ein und zeigt dabei folgende Grafik: Abbildung 3.1. Diese zeigt, wer welche Dokumente benötigt. Wobei die Dokumentationsgruppen lediglich Oberbegriffe sind. Detaillierte Informationen beinhalten folgende Werke [GBI<sup>+</sup>10, SE15].

	Modulsichten		C&C Sichten		Verteilung			Sonstige				
Rolle des Lesers	Dekomposition	Beziehungen(Use)	Generalisation	Schichten	Datenmodell	Verschiedene	Deployment	Implementation	Install	Interface	Context Diagrams	Dynamisches Verhalten
Künftiger Architect	d	d	d	d	d	d	d	s	d	d	d	d
SWPÄ	d	d	d	d	d	d	s	s		d	d	d
Entwickler anderer Systeme				s						d	o	
Betriebspersonal	s	s		s	s		d	d	o			

d = detailliert, s = einige details, o = übersicht, x = alles

Abbildung 3.1: Wer benötigt welche Art von Dokumentation [SE15]

Als kritisch wird im Rahmen des Fachartikels [SE15] sowie der Literatur [GBI<sup>+</sup>10] das nachträgliche Erstellen von Dokumentation erwähnt, denn dies ist nur durch Mitarbeiter der Entwicklungsteams möglich. Im Falle des nachträglichen Dokumentierens sollten laut ihr allerdings sog. Reengineering-Tools heran gezogen werden, welche bei diesem Prozess unterstützen. Außerdem bedeutet nachträgliches Dokumentieren einen enorm hohen Aufwand und sowie Kosten.

Abschließend erwähnt sie noch einmal den Wachstum, sowie die Komplexität neuer Softwaresysteme, dieser erfordert das große Ganze, die so genannte *Makro-Sicht* im Auge zu behalten und dementsprechend zu planen. Als Beispiel hierfür wird das Entwickeln in Baukastenform genannt, mit standardisierten Anschlüssen zur Kopplung verschiedener Businessprozesse. Als die wichtigsten Rollen sieht sie hierfür die Notationen *Business Process Modelling Language*, *Business Process Execution Language* und die dazugehörige Infrastruktur *Enterprise Service Bus* um Service Orientierte Systeme aufbauen zu können, welche laut ihr die Lösung auf die Vernetzung der Systeme sowie den wechselnden Anforderungen sind [SE15].

Eine weitere wichtige Schlussfolgerung von Frau Schreiber-Ehle ist die Notwendigkeit einer vertraglichen Regelung, welche Dokumentationen, in welcher Qualität und wann im Rahmen eines Softwareprojekts vom Auftragnehmer zu liefern sind, um etwaige Rechtsstreitereien zu vermeiden, sowie den Fall einer undokumentierten Software, welche ggf. wertlos sein kann [SE15].

## 3.2 Projektmanagement und die zugehörige Dokumentation

Der nächste im Rahmen dieser Arbeit verwendete Artikel ist von Wiegand Liesegang mit dem Titel *Projektmanagement und die zugehörige Dokumentation*, ebenso aus der Fachzeitschrift *Computer und Recht* [Lie15]. Der Autor versucht in seinem Artikel die Frage, welche Dokumentationsarten nach den allgemein anerkannten Regeln der Technik zu erstellen sind, zu beantworten. Im folgenden wird kurz erläutert was man unter den sog. *allgemein anerkannten Regeln der Technik* versteht, gefolgt von den Erkenntnissen Liesegangs im Bezug auf die Anforderungen an das Projektmanagement und die zugehörige Dokumentation in Kombination mit den sog. allgemein anerkannten Regeln der Technik.

### ***Allgemein anerkannte Regeln der Technik:***

Rein rechtlich ist diese Begrifflichkeit nirgends eindeutig definiert, allerdings ist man sich größtenteils einig darüber, dass etwas nach den allgemein anerkannten Regeln der Technik erfüllt ist, wenn eine technische Regel der Richtigkeitsüberzeugung der technischen Fachleute entspricht bzw. nach Stand der Wissenschaft für theoretisch richtig angesehen wird und in der Praxis erprobt bzw. bewährt ist [Lie15].

Um die allgemein anerkannten Regeln der Technik (aaRdT) im Bezug auf das Projektmanagement plus Dokumentation bestimmen zu können, greift Herr Liesegang auf die DIN 69901-2:2009 *Projektmanagement - Projektmanagementsysteme - Teil 2: Prozess, Prozessmodell* [Deu09a] zurück, welche als Basis dient. Dieser stellt er die zwei allgemeinen Projektmanagementmethoden PMOBK [20004] und PRINCE2 [Hin12] sowie die agile Methode Scrum [SS01] gegenüber. Dabei wird von ihm ein Mindeststandard in Form einer Gliederung von mehreren Prozessen aus DIN 69901-2 *Projektmanagement* [Deu09a] abgeleitet. Bezogen auf diesen Mindeststandard werden die Prozesse einander

gegenüber gestellt. Hierdurch ist gewährleistet, dass eine Bestimmung der aaRdT auf bewährte Praxis zurückgreift. Vorab werden die Methoden auf mögliche Eigenschaften zur Auswahl der Durchführungsmethode oder zur Ausgestaltung besagter Methoden geprüft. Zusammenfassend lässt sich hier sagen, dass die jeweiligen Methoden keine konkreten Aussagen darüber treffen, wann sie zu verwenden sind oder nicht, sondern vorab anhand bestimmter Kriterien entschieden werden muss, ob sich eine Methode überhaupt eignet. Des Weiteren besagen alle Methoden mit Ausnahme von Scrum, dass eine allgemein Projektmanagementdokumentation zu erstellen ist und wie diese auszusehen hat. Bei Scrum ist man dazu implizit verpflichtet. Daraus ergibt sich für aaRdT, dass es als bewährt erachtet wird, eine Dokumentation der Methode zu erstellen, diese beschreibt beispielsweise welche Prozesse im Projekt ausgeführt werden und alles was dazu gehört [Lie15].

Im Anschluss kommt Herr Liesegang zu den zuvor erwähnten Prozessen, diese stammen wie bereits erwähnt aus DIN 69901-2 [Deu09a]. Diesem werden dann die jeweiligen Pendant der anderen Methoden gegenüber gestellt. Als Ergebnis daraus folgt eine Aussage ob diese Mindestanforderung aus DIN 69901-2 [Deu09a] als anerkannte Regel der Technik gesehen werden kann bzw. was zusätzlich erfüllt sein muss. Folgende Punkte werden dabei auf Inhalt sowie Notwendigkeit einer zu erstellenden Dokumentation geprüft:

1. Ziele definieren
2. Projektstrukturplan erstellen
3. Ressourcenplan erstellen
4. Terminplan erstellen
5. Risiken analysieren
6. Gegenmaßnahmen zu Risiken planen
7. Zielerreichung steuern
8. Änderungen steuern
9. Ressourcen steuern
10. Termine steuern
11. Risiken steuern
12. Abnahme erteilen
13. Projekterfahrung sichern

Mit Ausnahme von Scrum gibt es für jede Methode ein äquivalentes Pendant, worin einerseits beschrieben wird, was zu tun ist und andererseits welche Dokumentationen erstellt werden müssen. Bei Scrum ist dies nicht an allen Stellen explizit definiert, implizit aber immer möglich. Somit ergibt sich, dass alle Punkte für die Praxis des Projektmanagements relevant sind und als bewährt einzustufen sind. Es kann aber sein, dass entsprechende technische Regeln mit einbezogen werden müssen. Ähnlich ist es bei der Projektmanagement-Dokumentation, diese ist entweder explizit zu erstellen oder es ist zumindest möglich. Dadurch ergibt sich ebenso, dass die Erstellung einer Dokumentation den allgemein anerkannten Regeln der Technik entspricht, ggf. unter Beachtung technischer Regeln. Dadurch folgt auch, dass sich in mehreren in der Praxis bereits bewährter Methoden etliche Gemeinsamkeiten erkennen lassen, lässt man Scrum außen vor, ist diese Menge noch größer [Lie15].

### 3.3 Software-Anforderungsdokumentation

Der dritte und letzte im Rahmen dieser Arbeit behandelte Artikel ist von Peter Hoppen mit dem Namen *Software-Anforderungsdokumentation* ebenso aus der Fachzeitschrift *Computer und Recht* [Hop15]. Er befasst sich innerhalb des Artikels mit der Anforderungsdokumentation einer Software, welche laut ihm, wenn nicht oder in mangelhafter Form vorliegend, ein Grund für das Scheitern eines IT-Projekts sein kann. Vorab erwähnt er hier eine Art Daumenregel, welche besagt, dass die Kosten rasant um das 10-fache ansteigen können, wenn eine Anforderung erst während der Entwicklung erkannt wird. Dies wäre vermeidbar, wenn die Anforderungen bereits in der Leistungsbeschreibung enthalten gewesen wären. Außerdem kann solch eine Dokumentation auch eine große Hilfe im Fall eines Rechtsstreits zwischen Auftraggeber und Auftragnehmer sein. Die Anforderungsdokumentation wird aus Sicht des Auftraggebers als eine Art Investitionsschutz gesehen, welche eine gewisse Qualität sicherstellen soll. [Hop15]

Als erstes folgt eine Definition der Anforderungen, dies beinhaltet die Erhebung sowie das Management der Anforderungen, z.B. welche Erhebungstechniken, Darstellungsformen, Prüftechniken etc. sollen verwendet werden sowie die Organisation von „Wissen“. Insgesamt ergeben sich drei Phasen zur Erstellung. Als erstes gilt es, sog. Stakeholder zu identifizieren, denn diese haben entweder direkten oder indirekten Einfluss auf die Anforderungen, somit müssen sie mit einbezogen werden. Als nächstes müssen Ziele sowie Rahmenbedingungen geschaffen und analysiert werden (Ziele der Stakeholder in Einklang miteinander gebracht und priorisiert werden). Rahmenbedingungen müssen erhoben werden, denn diese können den Lösungsraum deutlich einschränken. Zu den eben genannten Rahmenbedingungen gilt es die vorhandenen Dokumentationen ein zu fordern und zu analysieren. Final sind alle Anforderungen festzuhalten bzw. zu dokumentieren. [Hop15]

Des Weiteren gibt Herr Hoppen eine Definition darüber, welche Kriterien solch eine Dokumentation zu erfüllen hat. Sie soll beispielsweise äußerst umfassend sein, sich aber auf die wesentlichen Eigenschaften konzentrieren und eine vollständige Beschreibung der gewünschten Softwarelösung abgeben. Außerdem sollte sie dabei alle Bereiche abdecken. Der Detailgrad spielt hierbei eine essentielle Rolle, sowie die Vollständigkeit bzgl. Infrastruktur und Schnittstellen. Des Weiteren soll eine Anforderungsdokumentation selbstverständlich eindeutig, konsistent und widerspruchsfrei sein. Die von ihm gegebene Erläuterung lässt sich aus dem IEEE Standard 830 *Anforderungen an Anforderungen* [Iee98] ableiten. Wichtig ist außerdem, dass während dem gesamten Prozess auf eine einfache unmissverständliche Sprache geachtet wird, welche mit anschaulichen Grafiken ergänzt werden kann. Formulierungen sollten abschließend sein, d.h. die sog. Leistungsbeschreibung sollte von den Stakeholdern abgenommen (genehmigt) werden. Außerdem sollte es die Dokumentation ermöglichen, die Anforderungen im Projekt zu verfolgen, d.h. Prioritäten sollten vergeben werden, Abhängigkeiten erkennbar und verwaltet sein, etc. [Hop15].

Als nächstes folgt eine Beschreibung der technischen Inhalte einer Anforderungsdokumentation um eben genannte Punkte zu berücksichtigen. Als erstes sollte durch den Auftraggeber eine Leistungsbeschreibung, eine sog. ergebnisrelevante Anforderungsbeschreibung auch Lastenheft genannt, erstellt werden. Wichtig an dieser Stelle ist, dass der Auftraggeber hier nur das beschreibt, *was* um zu setzen ist und nicht *wie*. Lediglich einschränkende Rahmenbedingungen dürfen enthalten sein, welche ggf. vorgeben wie sich eine neue Softwarelösung o.ä. ein zu fügen hat. Das Problem an dieser Stelle

ist, dass dem Auftraggeber oft das nötige technische Wissen fehlt, um einschätzen zu können, was seine neue Software denn überhaupt können sollte, somit folgen auch oft Kommunikationsprobleme. [Hop15].

Daraufhin kommt der Auftragnehmer ins Spiel, welcher basierend auf dem Lastenheft ein sog. Pflichtenheft erstellt. Darin ist im Normalfall enthalten, *wie* er das gewünschte Produkt bzw. Funktionalität o.ä. umsetzen wird, d.h. es enthält die technischen Spezifikationen. Durch diese Leistungsdefinition versucht er auch seine Leistung einzugrenzen, da für manche Anforderungen der Aufwand schwer abzuschätzen ist und er im Normalfall bestrebt ist, mit Zusatzaufträgen in Form von Change Requests beauftragt zu werden. [Hop15].

Falls noch kein Lastenheft vorliegt, werden vom Auftragnehmer auch oft Abweichungsanalysen in Form von Workshops mit dem Auftraggeber durchgeführt. Hier soll ermittelt werden, falls bereits eine Standardlösung existiert, welche Funktionen zusätzlich entwickelt werden müssen. Auch das kann problematisch werden, z.B. fehlt dem Auftraggeber das grundlegende technische Verständnis oder ihm fehlt das nötige Wissen über das „Standardprodukt“ [Hop15]. Im nächsten Schritt gibt Herr Hoppen an, welche technischen Inhalte solch eine Beschreibung enthalten soll:

- Funktionale Anforderungen
  - Datenbestände -management
  - Berechtigungskonzept
  - Reportingfunktionen
  - Datenvolumen
  - Abnahmekriterien und Testfälle als Anforderungsspezifikation
  - Timing
- Qualitätsanforderungen
- Rahmenbedingungen
- Anforderungen an zu erstellende Dokumentation
  - Systemdokumentation und Installationsanweisung
  - Anwenderdokumentation
  - Vorgaben für die Systementwicklung
  - Quellcode-Dokumentation
  - Systemarchitektur und Basistechnologie
  - Datenmodell
  - Issue-Management und Leistungsaufzeichnung
  - Projekt- und Ergebnisdokumentation

Allgemein ist bei der Dokumentation der Leistung auf die funktionalen und nicht funktionalen Eigenschaften zu achten. Inhaltlich wird hier nicht genauer darauf eingegangen. Sie besteht letztendlich aus Software-Herstellungsdokumentation und Projektdokumentation. Des Weiteren wurde der Teil der *Anforderungen an zu erstellende Dokumentation* bereits in 3.1 bzw. von Frau Schreiber-Ehle [SE15] erläutert. [Hop15]

Bezogen auf die Form der Leistungsbeschreibung sagt Hoppen außerdem, dass sich formale Ansätze für die Beschreibung funktionaler Spezifikation nicht durchgesetzt haben, obwohl diese existieren würden. Dies führt dazu, dass Leistungsbeschreibungen ihr eigentliches Ziel verfehlen und für Stakeholder oft unverständlich bzw. nicht nachvollziehbar sind. [Hop15]

Des Weiteren gibt es verschiedene Ansätze bei der Erstellung einer Leistungsbeschreibung. Ein Beispiel wäre die textuelle Darstellung, welche auch die am häufigsten Verwendete ist, diese kann selbstverständlich wie bereits zuvor erwähnt von Grafiken oder Tabellen ergänzt werden. Das Problem an dieser Stelle ist aber ganz klar die nicht zu vermeidende Mehrdeutigkeit der natürlichen Sprache. Ein weiterer Ansatz ist eine schablonierte Darstellungsform, d.h. man bedient sich existierender Templates und passt diese an die eigenen Bedürfnisse an. Beispiele hierfür sind *PLanguage*, *Volere-Template*. Als besonders wichtig wird hier auch die Wiederverwendung der selben Begriffe für die selbe Bedeutung erwähnt, u.a. aus rechtlichen Gründen. Es gibt noch weitere Darstellungsformen wie *UML*, *User Stories*, *Geschäftsprozess-Darstellungen* usw. Hoppen geht außerdem stark davon aus, dass sich im Bereich systemunterstützter und formaler Strukturierung von Anforderungen in den nächsten Jahren einiges tun wird. Dies wiederum ist begründet durch offensichtliche Missstände, wie das nicht Vorhandensein anerkannter Regeln der Technik, sowie aktuellen Entwicklungen in diesem Bereich. [Hop15].

Ein weiterer wichtiger Punkt ist der Detailgrad der Anforderungen. In der Praxis fällt dieser nämlich meist zu gering aus, sodass bei der Abnahme bzw. den dazugehörigen Tests festgestellt wird, dass das Endprodukt eigentlich nicht die gewünschte Qualität erfüllt oder ggf. Funktionalitäten fehlen. Im Detail kann das verschiedene Ursachen haben, z.B. Anforderungen wurden nicht detailliert genug beschrieben oder erst zu spät berücksichtigt etc. Damit genau das nicht eintreten soll, muss von Anfang an versucht sein auf Seiten der Auftraggeber eine lösungsneutrale Problem- und Aufgabenstellung abzuliefern. Hier spielt der Detailgrad eine entscheidende Rolle, ist er zu gering, könnte beispielsweise oben genanntes passieren, ist er zu hoch, ist er meist zu lösungsorientiert und enthält bereits Informationen, *wie* etwas umgesetzt werden soll. Dies wiederum ist die eigentliche Aufgabe des Auftragnehmers. Dieser gibt anhand eines Pflichtenheftes an, *wie* er das *was* umsetzen möchte. Die Folge von dieser Missachtung wäre eine unnötige Einschränkung des Auftragnehmers, welche zu unnötigem Mehraufwand oder in extrem Fällen sogar dem falschen Produkt führt. [Hop15]

Letztendlich finden sich in allen möglichen Normen bzw. Durchführungsmethoden, Informationen, Vorgaben oder Mustergliederungen zu Leistungsbeschreibungen. Als Best Practice bezeichnet Hoppen PMBOK [20004] und SWEBOK [ABDM01].



## Kapitel 4

# Zusammenfassung

Im ersten Teil dieser Seminararbeit wurde eine grundlegende Einführung zum breiten Thema der Dokumentation in IT-Projekten gegeben. Daraufhin folgte die Zusammenfassung und Analyse mehrerer Fachartikel zum Thema Dokumentation. Wie der Titel bereits sagt, sollte im zweiten Teil besonders auf Neuerungen geachtet werden. Hier wurden von den einzelnen Spezialisten meist Missstände dargestellt und potentielle Lösungsvorschläge gegeben.

Frau Schreiber-Ehle welche sich im Rahmen ihres Artikels hauptsächlich mit der zu erstellenden Dokumentation während der Softwareerstellung beschäftigte, erörterte die Notwendigkeit verschiedener Dokumentationen in Abhängigkeit der Absichten bei der Verwendung oder Weiterentwicklung des Softwareprojekts nach Übergabe an den Auftraggeber, sowie den daran beteiligten Personen. Die Art sowie der Inhalt dieser Dokumente sind nicht neu und wurden bereits in der Literatur oder durch Normen festgehalten. Ein Aspekt im Rahmen ihres Artikels welcher neuartiger ist, ist die relevanter werdende Betrachtungsweise eines Softwareprojekts. Diese werden immer größer und komplexer, somit ist es laut der Autorin notwendig das Ganze aus einer sog. Makro-Sicht zu betrachten. Das bedeutet es soll versucht werden, weg zu kommen von der Ansicht des alleinstehenden Softwareprodukts. Dieses soll eher als Bestandteil eines großen Systems, in Baukastenform gesehen und geplant werden. Sodass dieses über standardisierte Anschlüsse verfügt, was bedeuten würde das Systeme verschiedener Businessprozesse miteinander gekoppelt werden können. Hierfür notwendige Notationsformen sind *Business Process Execution Language* und die dazugehörige Infrastruktur *Enterprise Service Bus*. Diese sind laut ihr Notwendig bei der Planung lokal und global vernetzter Systeme.

Herr Liesegang bezieht sich in seinem Artikel ausschließlich auf die Projektmanagement Dokumentation, in welcher er auf die gängige Praxis eingeht. Zudem bringt er aber auch für den Bereich Dokumentation einen neuwertigen Ansatz zur Sprache, die sog. allgemein anerkannten Regeln der Technik, welcher immer beliebter werden zu scheint. Wie vorhin bereits erläutert werden dabei mehrere Durchführungsstrategien gegenüber gestellt und auf bestimmte Eigenschaften geprüft. Letztendlich bedeutet dies, dass Elemente dieser Methoden kombiniert werden können, wenn sie die Bedingungen der aaRdT erfüllen.

Der letzte in dieser Arbeit behandelte Artikel von Herrn Hoppen befasst sich mit der Anforderungsdokumentation, welche von ihm als extrem wichtig erachtet

wird, er aber auch sagt, dass sie in der Praxis meist ihr Ziel verfehlt. Leistungsbeschreibungen seien beispielsweise oft ungenau bzw. lückenhaft und für Stakeholder nicht nachvollziehbar und dass oft etwas anderes umgesetzt wird, als von diesen erwartet. Außerdem geht Hoppen stark davon aus, dass bedingt durch Missstände wie den eben erwähnten, sowie Entwicklungen in diesem Bereich, sich in naher Zukunft einiges tun wird.

Artikelübergreifend lässt sich sagen, dass das Thema „Dokumentation“ in einem IT-Projekt hoch komplex werden kann bzw. ist, ihr in den meisten Fällen aber zu wenig Aufmerksamkeit geschenkt wird, aus Kosten- sowie Zeitgründen. Die Meinungen der zuvor erwähnten Fachleute, stellen diese Problematik dokumentationsartenübergreifend dar und geben Verbesserungsvorschläge. Außerdem wird von jedem der Drei auch die Wichtigkeit einer vertraglichen Regelung der Dokumentation erwähnt, da es nahezu keine gesetzliche Regelung gibt. Alle drei nutzen als Grundlage ihrer Argumentation sowohl Praxiserfahrung als auch bestehende Normen oder Fachliteratur. Die nennenswertesten neuen Fachmeinungen im Rahmen der drei Artikel, sind die Betrachtung der Makroschicht von Frau Schreiber-Ehle, der von Herrn Liesegang erwähnte Ansatz mit den allgemein anerkannten Regeln der Technik im Bereich der Projektmanagementdokumentation und der Äußerung von Herrn Hoppen, dass sich bzgl. systemunterstützter und formaler Strukturierung in naher Zukunft einiges tun wird, bedingt durch aktuelle Missstände in diesem Bereich

# Abbildungsverzeichnis

3.1	Wer benötigt welche Art von Dokumentation [SE15] . . . . .	8
-----	--	---



# Literaturverzeichnis

- [19512] *ISO-19505: Object Management Group Unified Modeling Language (OMG UML)*. ISO, 2012
- [19513] *ISO/IEC 19510:2013 - Information technology – Object Management Group Business Process Model and Notation*. ISO, 2013 [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=62652](http://www.iso.org/iso/catalogue_detail.htm?csnumber=62652)
- [20004] *A Guide To The Project Management Body Of Knowledge (PMBOK Guides)*. Project Management Institute, 2004. – ISBN 193069945X, 9781933890517
- [57111] IEEE Standard for Adoption of ISO/IEC 26514:2008 Systems and Software Engineering–Requirements for Designers and Developers of User Documentation. In: *IEEE Std 26514-2010* (2011), Jan, S. 1–72. <http://dx.doi.org/10.1109/IEEESTD.2011.5712775>. – DOI 10.1109/IEEESTD.2011.5712775
- [61011] ISO/IEC/IEEE Systems and software engineering – Content of life-cycle information products (documentation). In: *ISO/IEC/IEEE 15289:2011(E)* (2011), Nov, S. 1–94. <http://dx.doi.org/10.1109/IEEESTD.2011.6104079>. – DOI 10.1109/IEEESTD.2011.6104079
- [90096] *ISO 9000: quality management. ISO standards compendium; 6th ed.* Geneva : ISO, 1996 <https://cds.cern.ch/record/338013>
- [ABDM01] ABRAN, Alain (Hrsg.) ; BOURQUE, Pierre (Hrsg.) ; DUPUIS, Robert (Hrsg.) ; MOORE, James W. (Hrsg.): *Guide to the Software Engineering Body of Knowledge - SWEBOOK*. Piscataway, NJ, USA : IEEE Press, 2001. – ISBN 0769510000
- [bed] *Unvollständige Bedienungsanleitung als Sachmangel i.S.v. § 434 BGB*. [http://lorenz.userweb.mwn.de/urteile/olgmue6u4082\\_05.htm](http://lorenz.userweb.mwn.de/urteile/olgmue6u4082_05.htm), . – Accessed: 2016-12-27
- [Deu09a] DEUTSCHES INSTITUT FÜR NORMUNG E.V. (Hrsg.): *DIN 69901-2:2009-01 – Projektmanagement - Projektmanagementsysteme - Teil 2: Prozesse, Prozessmodell*. 9. Berlin, Wien, Zürich : Beuth Verlag GmbH, 2009 <http://www.beuth.de/en/standard/din-69901-5/113428752>
- [Deu09b] DEUTSCHES INSTITUT FÜR NORMUNG E.V. (Hrsg.): *DIN 69901-5:2009-01 – Projektmanagement – Projektmanagementsysteme – Teil 5: Begriffe*. 9. Berlin, Wien, Zürich : Beuth Verlag GmbH, 2009 <http://www.beuth.de/en/standard/din-69901-5/113428752>
- [din] *Programmstrukturen - DIN 66 261: Sinnbilder für Struktogramme nach Nassi-Shneiderman*. [https://web.archive.org/web/20070928094836/http://wwl.rh.fh-bielefeld.de/IN\\_Prak/inprak6.htm](https://web.archive.org/web/20070928094836/http://wwl.rh.fh-bielefeld.de/IN_Prak/inprak6.htm), . – Accessed: 2016-12-27

- [din81] *DIN 66230 : 1981-01 - Informationsverarbeitung; Programmdokumentation : mechanische Eigenschaften.* Berlin : Deutsches Institut für Normung e.V. -DIN- (Berlin) (Hrsg.); DIN (Hrsg.), 1981
- [dok] *Duden - Dokumentation.* <http://www.duden.de/rechtschreibung/Dokumentation>, . – Accessed: 2016-12-27
- [GBI<sup>+</sup>10] GARLAN, David ; BACHMANN, Felix ; IVERS, James ; STAFFORD, Judith ; BASS, Len ; CLEMENTS, Paul ; MERSON, Paulo: *Documenting Software Architectures: Views and Beyond.* 2nd. Addison-Wesley Professional, 2010. – ISBN 0321552687, 9780321552686
- [Her84] In: HERING, Ekbert: *Der Datenflußplan nach DIN 66001.* Wiesbaden : Vieweg+Teubner Verlag, 1984, S. 56–62
- [Hin12] HINDE, David: *PRINCE2 Study Guide.* 1st. Alameda, CA, USA : SYBEX Inc., 2012. – ISBN 1119970784, 9781119970781
- [Hop15] HOPPEN, Peter: Software-Anforderungsdokumentation. In: *Computer und Recht* (2015), Nr. 11, S. 747 – 760
- [Iee98] IEE, E.: IEEE Recommended Practice for Software Requirements Specifications. Version: Oktober 1998. <http://dx.doi.org/10.1109/ieeestd.1998.88286>. IEEE, Oktober 1998. – Forschungsbericht. – 1–40 S.
- [IEE08] IEEE: IEEE Standard for Software and System Test Documentation. In: *IEEE Std 829-2008* (2008)
- [Lie15] LIESEGANG, Wiegand: Projektmanagement und die dazugehörige Dokumentation. In: *Computer und Recht* (2015), Nr. 8, S. 541 – 556
- [RR09] REISS, M. ; REISS, G.: *Praxisbuch IT-Dokumentation.* Pearson Deutschland, 2009 <https://books.google.de/books?id=R0rXoYHEe94C>. – ISBN 9783827326812
- [SE15] SCHREIBER-EHLE, Sabine: Dokumentation in Softwareerstellungsverträgen. In: *Computer und Recht* (2015), Nr. 7, S. 469 – 481
- [Sof] *Softwaredokumentation.* <https://de.wikipedia.org/wiki/Softwaredokumentation>, . – Accessed: 2016-12-27
- [SS01] SCHWABER, Ken ; SUTHERLAND, Jeff: *The Scrum Guide.* 2001
- [Sta05] STANDARDIZATION, International O.: *ISO/IEC 19501: information technology - open distributed processing - unified modeling language (UML) version 1.4.2.* ISO, 2005 <https://books.google.de/books?id=2k-6kQEACAAJ>
- [sys] *Systemtechnische Dokumentation.* [http://www.jurpc.de/jurpc/show?id=iur\\_1988\\_0000\\_0010\\_0017\\_0018&type=pdf](http://www.jurpc.de/jurpc/show?id=iur_1988_0000_0010_0017_0018&type=pdf), . – Accessed: 2016-12-27
- [teca] *Technische Dokumentation.* [https://de.wikipedia.org/wiki/Technische\\_Dokumentation](https://de.wikipedia.org/wiki/Technische_Dokumentation), . – Accessed: 2016-12-27
- [tecb] *Was der Gesetzgeber sagt: Gesetze in der Technischen Dokumentation.* <http://www.commatec.de/en/ipa/info-portal/artikel/gesetze-normen-richtlinien.html>, . – Accessed: 2016-12-27
- [uml] *Unified Modeling Language.* <http://www.omg.org/spec/UML/>, . – Accessed: 2016-12-27