

Niklas Fischer

Immatrikulationsnummer.

Straße

PLZ Ort

E-Mailadresse

# Projektbericht

Thema:

Konzeption eines Lasten- und Pflichtenheftes für eine  
betriebsinterne Softwarelösung

*- öffentlich zugängliche Version -*

Betreuer:

Hamburg, 02.11.2024



## **Hinweis**

*Der folgende Text enthält Auszüge aus meinem Projektbericht.*

*Einige Abschnitte wurden entfernt, da sie vertrauliche Daten enthalten.*

*Bitte beachten Sie, dass die öffentliche Version von der eingereichten Arbeit teils stark abweicht.*

*Der Projektbericht wurde mit der Note sehr gut bewertet.*

## Inhaltsverzeichnis

I. Abbildungsverzeichnis.....	II
II. Tabellenverzeichnis.....	III
III. Abkürzungsverzeichnis.....	IV
<b>1. Einleitung</b> .....	<b>1</b>
1.1 Relevanz des Themas.....	1
1.2 Ziele der Arbeit.....	1
1.3 Aufbau der Arbeit.....	1
<b>2. Grundlagen</b> .....	<b>2</b>
2.1 Definition Lastenheft und Pflichtenheft.....	2
2.2 Aufbau von Lasten- und Pflichtenheften.....	6
2.3 Bedeutung des Lasten- und Pflichtenheftes im Softwareentwicklungsprozess.....	8
2.3.1 Wasserfallmodell.....	9
2.3.2 V-Modell (XT).....	10
2.3.3 Agile Vorgehensmodelle (Scrum).....	12
2.4 Definition und Bedeutung von Materialzertifikaten.....	13
<b>3. Hauptteil</b> .....	<b>14</b>
3.1 Konzeption des Lastenheftes.....	14
3.1.1 Das Unternehmen.....	14
3.1.2 Ausgangssituation.....	14
3.1.3 Ziele des Projekts.....	14
3.1.4 Anforderungen.....	14
3.1.4.1 Funktionale Anforderungen.....	14
3.1.4.2 Nicht-funktionale Anforderungen.....	15
3.1.5 Bestehende Architektur und Definition von Schnittstellen.....	15
3.2 Konzeption des Pflichtenheftes.....	15
3.2.1 Soll-Zustand.....	15
3.2.2 Systemarchitektur.....	15
3.2.3 Benutzeroberflächen.....	15
3.2.4 Grobe Projektplanung.....	15
3.2.5 Weiterentwicklung und mögliche Anpassungen.....	15
<b>4. Schlussbetrachtung</b> .....	<b>15</b>
IV. Literaturverzeichnis - Buchquellen.....	VI
V. Literaturverzeichnis - Internetquellen.....	IX

## **I. Abbildungsverzeichnis**

Abb. 1: Softwareentwicklungsphasen (links) und ihre Ergebnisse (rechts) am Beispiel des klassischen Wasserfallmodells .....	10
Abb. 2: V-Modell .....	11

## II. Tabellenverzeichnis

Tab. 1: Definitionen Lastenheft (eigene Darstellung) .....	3
Tab. 2: Definitionen Pflichtenheft (eigene Darstellung) .....	5
Tab. 3: Verschiedene Herangehensweisen zum Aufbau eines Lastenhefts (eigene Darstellung) ..	7
Tab. 4: Verschiedene Herangehensweisen für den Aufbau eines Pflichtenhefts (eigene Darstellung) .....	8

### **III. Abkürzungsverzeichnis**

VDI            Verein Deutscher Ingenieure

## **1. Einleitung**

### **1.1 Relevanz des Themas**

Die fortschreitende digitale Transformation bietet Unternehmen vielfältige Möglichkeiten, um betriebliche Prozesse effizienter zu gestalten und sich langfristig Wettbewerbsvorteile zu sichern. Ein wesentliches Ziel ist es, durch den Einsatz moderner Softwarelösungen Kosten zu optimieren, Prozesszeiten zu verkürzen und menschliche Fehler zu minimieren.

Der Schwerpunkt liegt dabei auf Prozessen, die sich aufgrund ihres repetitiven Charakters, ihrer klar definierten Abläufe und Regeln besonders gut für eine Automatisierung eignen.

Die erfolgreiche Integration dieser Softwarelösungen in bestehende Strukturen stellt jedoch für zahlreiche Unternehmen eine Herausforderung dar.

Softwareentwicklungsprojekte sind zudem häufig mit hohen Kosten verbunden. Ohne adäquate Planungsinstrumente besteht das Risiko, dass Zeitrahmen und Budgets schnell überschritten werden. Um solche Risiken zu minimieren, bieten Lasten- und Pflichtenhefte als bewährte Werkzeuge der Softwareentwicklung einen strukturierten Ansatz.<sup>1</sup>

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

### **1.2 Ziele der Arbeit**

Hauptziel der vorliegenden Arbeit ist die Konzeption eines Lasten- und Pflichtenheftes für eine unternehmensinterne Softwarelösung im Bereich Qualitätsmanagement, das als Planungsgrundlage für die erfolgreiche Umsetzung der Maßnahmen dient.

Zwischenziele sind die Definition der Begriffe Lastenheft und Pflichtenheft sowie die Vorstellung verschiedener Ansätze zum Aufbau dieser Dokumente aus der Literatur. Die Einordnung dieser Artefakte in den Softwareentwicklungsprozess stellt ein weiteres Zwischenziel dar.

### **1.3 Aufbau der Arbeit**

In den Grundlagen dieser Arbeit werden zunächst in Kapitel 2.1 die Begriffe Lastenheft und Pflichtenheft anhand verschiedener Definitionen aus der Literatur definiert. In Kapitel 2.2 werden unterschiedliche Herangehensweisen zum Aufbau dieser Dokumente analysiert.

---

<sup>1</sup> Vgl. Madauss (2020), S. 302 ff.; vgl. Broy und Kuhrmann (2021), S. 9

Anschließend wird in Kapitel 2.3 die Bedeutung von Lasten- und Pflichtenheften im Kontext des Softwareentwicklungsprozesses betrachtet. Dazu werden verschiedene Vorgehensmodelle erläutert und die Artefakte in die Modelle eingeordnet. Im Mittelpunkt steht dabei die Frage, ob Lasten- und Pflichtenhefte im agilen Zeitalter noch Relevanz haben.

Da Materialzertifikate in dieser Arbeit eine zentrale Rolle spielen, wird abschließend in Kapitel 2.4 auf die Definition und Bedeutung von Materialzertifikaten für den betrieblichen Qualitätsmanagementprozess eingegangen.

Auf Basis der gewonnenen Erkenntnisse wird im Hauptteil der Arbeit zunächst das Lastenheft erstellt. Zunächst erfolgt eine Darstellung des Unternehmens (Kapitel 3.1.1) und eine Analyse der Ausgangssituation (Kapitel 3.1.2).

Daraufhin werden in Kapitel 3.1.3 die Ziele des Projekts definiert und sowohl die funktionalen als auch die nicht-funktionalen Anforderungen an das System herausgearbeitet (Kapitel 3.1.4). Im letzten Teil des Lastenheftes werden die bestehende Systemarchitektur und die Schnittstellen zwischen den Systemkomponenten beschrieben.

Die dargestellten Aspekte des Lastenheftes werden anschließend in der Konzeption des Pflichtenheftes aufgenommen. In Kapitel 3.2.1 wird der angestrebte Soll-Zustand vorgestellt. Die neue Systemarchitektur wird in Kapitel 3.2.2 beschrieben. Die Benutzeroberflächen werden in Kapitel 3.2.3 erläutert. Darauf folgt der grobe Projektplan in Kapitel 3.2.4. Abschließend werden in Kapitel 3.2.5 Aspekte der Weiterentwicklung und mögliche Anpassungen behandelt.

## **2. Grundlagen**

### **2.1 Definition Lastenheft und Pflichtenheft**

In der Literatur wird bei der Definition des Begriffs Lastenheft häufig auf die Norm DIN 69901-5 zurückgegriffen, nach der ein Lastenheft die „vom Auftraggeber festgelegte Gesamtheit der Forderungen an die Lieferung und Leistung eines Auftragnehmers innerhalb eines (Projekt-) Auftrags“<sup>2</sup> beschreibt.

Ein Großteil der Definitionen (siehe Tab. 1) betont die fachliche Sicht des Auftraggebers. Aufgabe des Auftraggebers ist es, die Anforderungen aus Nutzersicht zu dokumentieren und

---

<sup>2</sup> DIN 69901-5 (2009), Kapitel 3.32

alle Randbedingungen zu erfassen. Verfügt der Auftraggeber nicht über das notwendige Fachwissen, sollte er jemanden mit dieser Aufgabe beauftragen.<sup>3</sup> Auf dieser Basis beauftragt der Auftraggeber „den Auftragnehmer mit der Realisierung der Anforderungen.“<sup>4</sup>

VDI (2014)	„Vom Auftraggeber oder in dessen Auftrag erstellte Zusammenstellung aller Anforderungen des Auftraggebers hinsichtlich Liefer- und Leistungsumfang als Ausschreibungs-, Angebots- und/oder Vertragsgrundlage.“ <sup>5</sup>
Hahn (2020)	„Das Lastenheft, manchmal auch als Grobkonzept bezeichnet, definiert das Was und das Wofür. Es listet all jene Details auf, die der Auftraggeber als Zielbestimmungen für das Projekt festlegt.“ <sup>6</sup>
Lettau (2000)	„Das Lastenheft des Auftraggebers beschreibt die Zielsetzungen, Aufgabenstellungen und Eckdaten des Webprojekts und bedient sich dabei der Dokumentation des Istzustands mit anschließender Erläuterung des Sollzustands.“ <sup>7</sup>

Tab. 1: Definitionen Lastenheft (eigene Darstellung)

Grundsätzlich definiert das Lastenheft den Funktionsinhalt eines Projektes, also das "WAS" und "WOFÜR" und beschreibt das Problem aus Anwendersicht.

Einerseits dient es als Ausschreibungsdokument, auf dessen Grundlage sich potenzielle Auftragnehmer bewerben können. Andererseits stellt es auch ein wichtiges Vertragsdokument dar, in dem die Erwartungen und Anforderungen klar festgelegt werden.<sup>8</sup>

Des Weiteren dient es als zentrale Planungsgrundlage für die Projektleitung, als Leitfaden für alle involvierten Gruppen sowie als Instrument zur Überwachung des Projektfortschritts.<sup>9</sup>

Lastenhefte finden sowohl bei der Ausschreibung zur Erstellung von Individualsoftware als auch bei der Auswahl und Beschaffung von bereits existierender Standardsoftware Anwendung. Im Rahmen dessen ist eine Kategorisierung der Anforderungen in "kritisch", "gefordert" und

<sup>3</sup> Vgl. Krypczyk (2018), S. 222

<sup>4</sup> Hammerschall und Beneken (2013), S. 144

<sup>5</sup> VDI (2014), S. 3

<sup>6</sup> Hahn (2020), S. 53

<sup>7</sup> Lettau (2000), S. 75

<sup>8</sup> Vgl. VDI (2001), S. 2; vgl. Osterhage (2016), S. 157; vgl. Broy und Kuhrmann (2021), S. 257

<sup>9</sup> Vgl. Stender (2011), S. 21

"optional" empfehlenswert. Die als kritisch eingestuften Anforderungen sind für das Projekt unverzichtbar. Sollte die Softwarelösung eines Anbieters diese K.O.-Kriterien nicht erfüllen, muss dies unmittelbar bei der Analyse festgestellt werden.

Geforderte Anforderungen sind notwendig, um die Projektziele zu erreichen, während optionale Anforderungen als Bonus betrachtet werden.

Diese optionalen Anforderungen ermöglichen es zu prüfen, ob die neue Lösung zusätzliche Funktionen bietet. Bei der Bewertung sollte insbesondere geprüft werden, ob der Nutzen der zusätzlichen Funktionen die damit verbundenen Kosten rechtfertigt und ob diese Funktionen tatsächlich erforderlich sind.<sup>10</sup>

Auf der Grundlage der im Lastenheft definierten Anforderungen erstellt der Auftragnehmer das Pflichtenheft, in dem detailliert beschrieben wird, wie der gewünschte Soll-Zustand unter Berücksichtigung konkreter Lösungsansätze erreicht werden soll.<sup>11</sup>

Neben dem Begriff "Pflichtenheft" finden sich in der Praxis, häufig auch andere Bezeichnungen wie "Feinkonzept", "Fachspezifikation" oder "Sollkonzept".<sup>12</sup>

Das Pflichtenheft ist in der Regel Bestandteil des Angebots des Auftragnehmers.<sup>13</sup>

Der Auftraggeber ist dafür verantwortlich, die Widerspruchsfreiheit und Umsetzbarkeit der im Lastenheft definierten Anforderungen zu überprüfen.

Nach Genehmigung des Pflichtenheftes durch den Auftraggeber wird das Pflichtenheft Vertragsbestandteil.<sup>14</sup>

Anschließend dient es als Referenzdokument, um zu überprüfen, ob die erbrachte Leistung vollständig und gemäß den definierten Anforderungen umgesetzt wurde.<sup>15</sup>

DIN 69901-5 (2009)	Das Pflichtenheft enthält die vom „vom Auftragnehmer erarbeitete Realisierungsvorgabe auf der Basis des vom Auftraggeber vorgegebenen Lastenheftes.“ <sup>16</sup>
-----------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------

<sup>10</sup> Vgl. Groß und Pfenning (2019), S. 239; vgl. Bär et al. (2017), S. 13 f.

<sup>11</sup> Vgl. Lettau (2000), S. 146; vgl. Osterhage (2016), S. 157

<sup>12</sup> Vgl. Bär et al. (2017), S. 14

<sup>13</sup> Vgl. Alpar et al. (2023), S. 407 f.

<sup>14</sup> Vgl. VDI (2001), S. 3; vgl. Teich et al. (2008), S. 55; vgl. Brandt-Pook und Kollmeier (2020), S. 13

<sup>15</sup> Vgl. Eisele et al. (2023) S. 122; vgl. VDI (2001), S. 4

<sup>16</sup> DIN 69901-5 (2009), Kapitel 3.40

VDI (2001)	„Beschreibung der Realisierung aller Anforderungen des Lastenheftes. [...] Im Pflichtenheft wird definiert WIE und WOMIT die Anforderungen zu realisieren sind.“ <sup>17</sup>
Hahn (2020)	„Das Pflichtenheft, manchmal auch als Feinkonzept bezeichnet, definiert das Wie und das Womit.“ <sup>18</sup>
Lettau (2000)	„Ein Pflichtenheft beschreibt die Herangehensweise, das zu verwendende Web-Konzept und damit den Lösungsweg des Auftragnehmers zur Umsetzung eines Webprojekts. Zentrales Element ist dabei das ausgewogene Zusammenspiel von Informations-, Navigationsarchitektur und Layout-Design.“ <sup>19</sup>

Tab. 2: Definitionen Pflichtenheft (eigene Darstellung)

Die zuvor dargestellten Definitionen (s.o.) betonen eine klare Abgrenzung zwischen dem Lastenheft, das vom Auftraggeber erstellt wird und ausschließlich der Beschreibung des Problems dient, und dem Pflichtenheft, das vom Auftragnehmer erstellt wird und sich auf die Beschreibung des Lösungsraumes konzentriert.

In der Praxis lässt sich jedoch einerseits „die klare Trennung von Problem- und Lösungsraum mithilfe der beiden Dokumente“<sup>20</sup> nicht vollständig gewährleisten. Andererseits ist es auch möglich, dass der Auftragnehmer den Auftraggeber bei der Erstellung des Lastenheftes unterstützt, insbesondere dann, wenn der Auftraggeber nicht über die erforderliche methodische und fachliche Kompetenz verfügt.

Ebenso ist es möglich, dass der Auftraggeber gemeinsam mit dem Dienstleister das Pflichtenheft ausarbeitet, wenn dieser Einfluss auf die Lösung nehmen möchte.<sup>21</sup>

Die Bedeutung von Lasten- und Pflichtenheften ist nicht zu unterschätzen. Diesbezüglich sei auf die Erkenntnisse verschiedener Autoren verwiesen, darunter Teich et al. (2008), Groß und Pfenning (2019), Broy und Kuhrmann (2021) sowie Madauss (2020), die übereinstimmend

<sup>17</sup> VDI (2001), S. 2 f.

<sup>18</sup> Hahn (2020), S. 58

<sup>19</sup> Lettau (2000), S. 146

<sup>20</sup> Hammerschall und Beneken (2013), S. 147

<sup>21</sup> Vgl. Hammerschall und Beneken (2013), S. 147

betonen, dass sie eine entscheidende Rolle für den langfristigen Erfolg von Softwareprojekten spielen.<sup>22</sup>

Lasten- und Pflichtenhefte werden im Rahmen von Projekten in vielen Branchen – auch außerhalb der Softwareentwicklung - eingesetzt, wobei sich in einigen Bereichen standardisierte Vorgaben etabliert haben. International wird für die Softwareentwicklung häufig auf die Norm ISO/IEC/IEEE 29148:2018 verwiesen. Diese Norm unterscheidet nicht explizit zwischen dem Lasten- und Pflichtenheft und behandelt Aspekte des Anforderungsmanagements wesentlich detaillierter.<sup>23</sup> Am ehesten vergleichbar mit den traditionellen Lasten- und Pflichtenheften ist das Kapitel 9.6, das sich mit der "Software Requirements Specification" befasst.<sup>24</sup>

Die Praxis hat aber gezeigt, dass diese Vorgehensweise ab einer bestimmten Projektgröße insbesondere aus rechtlicher Sicht nicht zu empfehlen ist.<sup>25</sup>

## 2.2 Aufbau von Lasten- und Pflichtenheften

Die häufig verwendete DIN-Norm liefert keine spezifischen Vorgaben zur Gliederung und zu den Inhalten des Lasten- und Pflichtenheftes. Andere Autoren und Normen, insbesondere Balzert (2009), Broy (2021) und der VDI (2014) bieten hingegen Vorschläge zum strukturellen Aufbau und zu den erforderlichen Inhalten.

Balzert (2009)	VDI (2014)	Broy und Kuhrmann (2021)
1. Visionen und Ziele 2. Rahmenbedingungen 3. Kontext und Überblick	1. Einführung in das Projekt 2. Beschreibung der Ausgangssituation (Istzustand) 3. Aufgabenstellung (Soll-Zustand)	1. Festlegung des Produktziels 2. Festlegung der Produktumgebung 3. Festlegung des Produkteinsatzes

<sup>22</sup> Vgl. Teich et al. (2008), S. 55; vgl. Groß und Pfenning (2019), S. 537; vgl. Broy und Kuhrmann (2021) S. 257; vgl. Madauss (2020), S. 110

<sup>23</sup> Vgl. Krypczyk und Bochkor (2018), S. 222; vgl. Stender, P. (2011), S. 21

<sup>24</sup> Vgl. ISO/IEC/IEEE (2018), Kapitel 9.6

<sup>25</sup> Vgl. Krypczyk und Bochkor (2018), S. 235

4. Funktionale Anforderungen	4. Anforderung an die Kommunikations-schnittstellen	4. Festlegung des Datenmodells
5. Qualitätsanforderungen	5. Anforderungen an die Systemtechnik	5. Festlegung der Funktionalität
	6. Anforderungen für Systementwicklung, -inbetriebnahme und -einsatz	6. Festlegung des Risikopotenzials
	7. Anforderungen an die Qualität	
	8. Anforderungen an die Projektabwicklung	

Tab. 3: Verschiedene Herangehensweisen zum Aufbau eines Lastenhefts (eigene Darstellung)<sup>26</sup>

In den verschiedenen Ansätzen zum Aufbau eines Lastenheftes lässt sich eine gemeinsame Struktur erkennen. In allen vorgestellten Herangehensweisen findet sich eine Beschreibung der Ausgangssituation, eine Definition der Projektziele sowie die Erfassung spezifischer Anforderungen. Auffallend ist jedoch, dass die Projektplanung bei Balzert (2009) nicht explizit behandelt wird, während Broy (2021) zusätzlich das Risikopotenzial berücksichtigt.

Analog zum Lastenheft existieren auch für das Pflichtenheft diverse Vorschläge hinsichtlich des Aufbaus. Die verschiedenen Ansätze (s.u.) zeichnen sich durch signifikante strukturelle Unterschiede aus. Balzert definiert das Pflichtenheft als „eine Verfeinerung und Präzisierung des Lastenhefts“<sup>27</sup> und folgt dabei einem Aufbau, der eng an das Lastenheft angelehnt ist. Im Kontrast dazu präsentiert der VDI einen deutlich flexibleren Ansatz, der eine offenerere Gestaltung ermöglicht. Auch Broy orientiert sich in seinem Vorschlag an der Struktur des Lastenhefts, wobei die Kapitel des Pflichtenhefts als direkte Antwort auf die im Lastenheft definierten Kapitel zu verstehen sind.

Balzert misst der Projektplanung keine Bedeutung bei, während in der VDI-Norm die

<sup>26</sup> Vgl. Balzert (2009), S. 494; vgl. VDI (2014), S. 1 ff.; vgl. Broy und Kuhrmann (2021), S. 257

<sup>27</sup> Balzert (2009), S. 495

Anforderungen an die Projektabwicklung im Lastenheft festgelegt werden.

Bei Broy und Kuhmann hingegen wird der Projektplan im Pflichtenheft definiert.

Balzert (2009)	VDI (2014)	Broy und Kuhmann (2021)
<ol style="list-style-type: none"> <li>1. Visionen und Ziele</li> <li>2. Rahmenbedingungen</li> <li>3. Kontext und Überblick</li> <li>4. Funktionale Anforderungen</li> <li>5. Qualitätsanforderungen</li> <li>6. Abnahmekriterien</li> <li>7. Subsystemstruktur (optional)</li> </ol>	<ol style="list-style-type: none"> <li>1. Systemtechnische Lösungen</li> <li>2. Systemtechnik (Ausprägung)</li> </ol>	<ol style="list-style-type: none"> <li>1. Abgrenzung des Aufgabenbereichs</li> <li>2. Festlegung der Systemschnittstellen</li> <li>3. Festlegung der Produktfunktionen</li> <li>4. Festlegung der Leistungsmerkmale</li> <li>5. Festlegung der Benutzungsschnittstelle</li> <li>6. Festlegung des groben Projektplans</li> <li>7. Festlegung des Risikopotenzials</li> </ol>

Tab. 4: Verschiedene Herangehensweisen für den Aufbau eines Pflichtenhefts (eigene Darstellung)<sup>28</sup>

Trotz fehlender Standardisierung sollte das Pflichtenheft, als Antwort auf das Lastenheft, dessen Anforderungen und Probleme aufgreifen und konkrete Anwendungsfälle beschreiben. Im IT-Kontext zählen unter anderem Design-Layouts und die Informationsarchitektur dazu.<sup>29</sup>

### 2.3 Bedeutung des Lasten- und Pflichtenheftes im Softwareentwicklungsprozess

Die Vielzahl an unterschiedlichen Vorgehensmodellen zur Umsetzung von IT-Projekten sowie die variierenden Ansätze im Umgang mit Lasten- und Pflichtenheften machen eine genauere Betrachtung erforderlich. Gemäß Broy und Kuhmann (2013) beschreibt ein Vorgehensmodell

<sup>28</sup> Vgl. Balzert (2009), S. 495 f.; vgl. VDI (2014), S. 1 ff.; Broy und Kuhmann (2021), S. 374

<sup>29</sup> Vgl. Krypczyk und Bochkor (2018), S. 226; vgl. Bär et al. (2017), S. 14; Lettau (2000), S. 146

„systematische, organisatorische, ingenieurmäßige und quantifizierbare Vorgehensweisen, um Aufgaben einer bestimmten Klasse wiederholbar zu lösen.“<sup>30</sup>

Ein Vorgehensmodell definiert den organisatorischen Rahmen für ein Projekt, dessen Ziel die Entwicklung eines Softwaresystems ist, und beeinflusst maßgeblich die Art und Weise, wie Anforderungen erfasst, dokumentiert und umgesetzt werden.<sup>31</sup>

Zu ihnen zählen klassische Ansätze, wie das Wasserfallmodell und das V-Modell mit der Erweiterung XT als auch moderne Ansätze wie Scrum.

### **2.3.1 Wasserfallmodell**

Das Wasserfallmodell ist „das älteste und bekannteste Vorgehensmodell für IT-Projekte, insbesondere für die Softwareentwicklung“<sup>32</sup>, welches durch Royce (1970) und später Boehm (1981) an Bedeutung erlangte. Die einzelnen Phasen des Projekts – von der Anforderungsanalyse über Entwurf und Implementierung bis hin zu Test und Wartung – sind klar voneinander abgegrenzt und werden nacheinander durchlaufen. Jede Phase endet mit einer Qualitätsprüfung, die sicherstellt, dass die Anforderungen korrekt umgesetzt wurden, bevor der nächste Schritt eingeleitet wird. Während Rückkopplungen zwischen unmittelbar benachbarten Phasen möglich sind, erfordern Rücksprünge zu früheren Phasen aufgrund von Fehlern oder geänderten Anforderungen stets eine neue Qualitätsprüfung.<sup>33</sup>

---

<sup>30</sup> Broy und Kuhrmann (2013), S. 86

<sup>31</sup> Vgl. Goll (2011), S. 68

<sup>32</sup> Kaufmann und Mülder (2023), S. 434

<sup>33</sup> Vgl. Broy und Kuhrmann (2021), S. 86 f.; vgl. Ernst et al. (2023), S. 724 f.

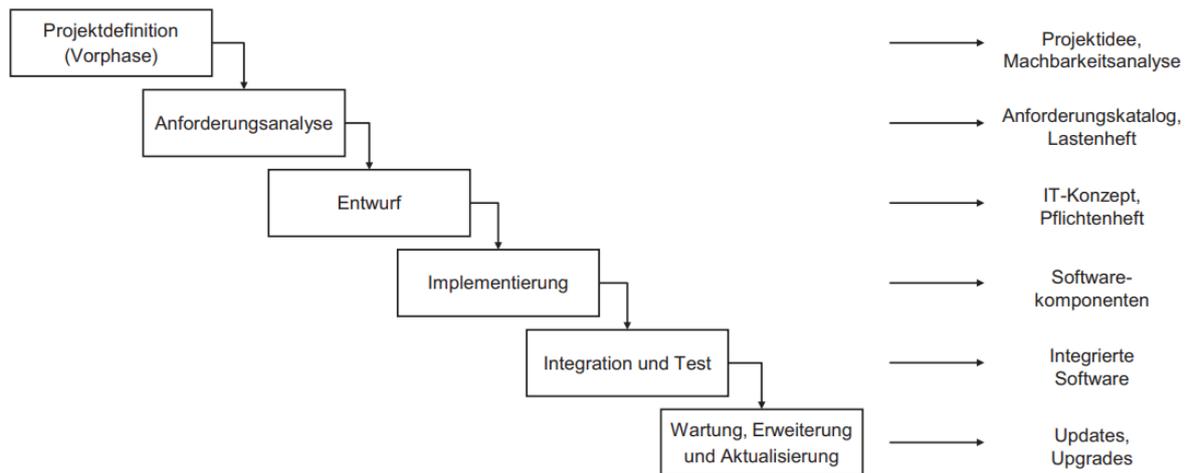


Abb. 1: Softwareentwicklungsphasen (links) und ihre Ergebnisse (rechts) am Beispiel des klassischen Wasserfallmodells<sup>34</sup>

Das Lastenheft ist das Ergebnis der Anforderungsanalyse im Wasserfallmodell, während das Pflichtenheft im nachfolgenden Schritt "Entwurf" erstellt wird.<sup>35</sup>

Die mangelnde Flexibilität sowie die späte Einbindung des Kunden im Wasserfallmodell führten zur Entwicklung alternativer Vorgehensmodelle.<sup>36</sup>

### 2.3.2 V-Modell (XT)

Das V-Modell, welches „das erstmals in den 1980er Jahren beschrieben wurde“<sup>37</sup>, wird häufig als Erweiterung des klassischen Wasserfallmodells angesehen.

Im Gegensatz zum rein sequenziellen Ansatz des Wasserfallmodells, das lineare Entwicklungsschritte vorschreibt, verfolgt das V-Modell einen systematischeren und stärker strukturierteren Ansatz, der die Gegenüberstellung zwischen Entwicklung und Qualitätssicherung betont.<sup>38</sup>

Der linke Zweig des V-Modells (Abb. 2) beschreibt die verschiedenen Phasen der System- und Softwareentwicklung, beginnend mit der Anforderungsanalyse bis hin zur Implementierung. Der rechte Zweig des Modells konzentriert sich auf die entsprechenden Qualitätsmaßnahmen, die parallel zu den Entwicklungsphasen ablaufen. Dies bedeutet, dass jeder Entwicklungsphase

<sup>34</sup> Alpar et al. (2023), S. 396

<sup>35</sup> Vgl. Alpar et al. (2023), S. 396

<sup>36</sup> Vgl. Ernst et al. (2023), S. 725

<sup>37</sup> Brandt-Pook und Kollmeier (2020), S. 25

<sup>38</sup> Vgl. Brandt-Pook und Kollmeier (2020), S. 25; vgl. Broy und Kuhmann (2021), S. 88

eine zugehörige Testphase zugeordnet ist, um sicherzustellen, dass die entwickelten Komponenten den Anforderungen entsprechen und korrekt implementiert sind.<sup>39</sup>

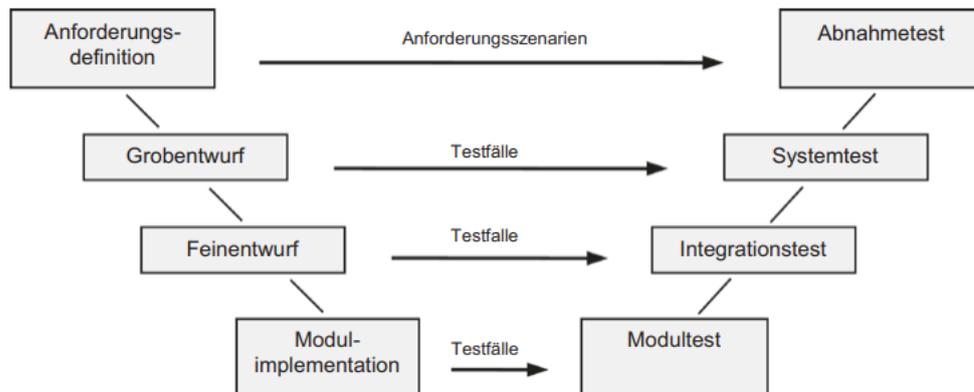


Abb. 2: V-Modell<sup>40</sup>

Das Lastenheft wird in der Phase Anforderungsdefinition erstellt, während das Pflichtenheft der Phase Grob- bzw. Feinentwurf zugeordnet wird.<sup>41</sup>

Das ursprüngliche V-Modell wurde im Laufe der Jahre kontinuierlich weiterentwickelt, um den Anforderungen moderner Softwareentwicklungsprojekte gerecht zu werden. Eine entscheidende Neuerung stellt dabei das V-Modell XT dar, welches für "eXtreme Tailoring" steht und die flexible Anpassung von Projekten an individuelle Anforderungen in den Vordergrund stellt.<sup>42</sup>

Durch einen modularen Aufbau in den Bereichen Produktmodell, Rollenmodell und Prozessabläufe ermöglicht es den „Ein- und Ausbau von Vorgehensbausteinen für ein spezifisches Projekt“<sup>43</sup>. Zudem erlaubt es im Gegensatz zum stark sequenziell geprägten klassischen V-Modell die Integration agiler Methoden.<sup>44</sup>

<sup>39</sup> Vgl. Duschl (2024), S. 148; vgl. Brandt-Pook und Kollmeier (2020) S. 25; vgl. Broy und Kuhrmann (2021), S. 88

<sup>40</sup> Brandt-Pook und Kollmeier (2020), S. 25

<sup>41</sup> Vgl. Broy und Kuhrmann (2021), S. 101

<sup>42</sup> Vgl. Duschl (2024), S. 149

<sup>43</sup> Metzner (2020), S. 23

<sup>44</sup> Vgl. Metzner (2020), S. 22

### 2.3.3 Agile Vorgehensmodelle (Scrum)

Mit der wachsenden Komplexität moderner Softwareprojekte stoßen traditionelle Ansätze wie die Wasserfallmethode oder das ursprüngliche V-Modell zunehmend an ihre Grenzen. Diese klassischen Modelle basieren auf starren, sequenziellen Abläufen und umfangreichen Vorgaben, die in dynamischen und schnelllebigen Projektumgebungen nicht immer flexibel genug sind. In den 1990er Jahren formierte sich in der Softwareentwicklung eine Gegenbewegung, die 2001 mit der Veröffentlichung des sogenannten "Agilen Manifests" durch eine Gruppe führender Softwareentwickler entscheidende Bekanntheit erlangte.

Während zuvor Prozesse, Werkzeuge und detaillierte Pläne im Fokus standen, rückt das Agile Manifest Flexibilität, Kundeninteraktion und die Fähigkeit, auf Veränderungen zu reagieren, in den Vordergrund.<sup>45</sup>

Scrum hat sich als die am weitesten verbreitete Methode der agilen Softwareentwicklung etabliert. Laut einer Studie von VersionOne und Collab.net steht „keine andere Methodik [...] so sehr für das agile Arbeiten und wird so häufig genutzt wie die Methode "SCRUM".“<sup>46</sup> Agilität selbst wird dabei allgemein als die Fähigkeit verstanden, Veränderungen offen zu begegnen und deren Auswirkungen effektiv zu bewältigen.<sup>47</sup>

Scrum zeichnet sich dabei durch besonders wenige Vorgaben in Bezug auf Vorgehensweise, Dokumentation und Testen aus und geht damit noch einen Schritt weiter als die allgemeinen Prinzipien der agilen Entwicklung.

Im Mittelpunkt stehen klar definierte Rollen (Product Owner, Scrum Master und Team), eine spezielle Besprechungsphilosophie und festgelegte Entwicklungszyklen (Sprints) sowie Artefakte (Produkt-Backlog und Sprint-Backlog).<sup>48</sup>

Bei der Anwendung agiler Methoden wie Scrum oder Kanban wird zwar auf die Erstellung eines expliziten Lasten- und Pflichtenheftes verzichtet, wie dies bei traditionellen Vorgehensmodellen üblich ist. Dennoch bleibt die Bedeutung einer umfassenden Anforderungsdokumentation bestehen. Auch in agilen Projekten zeigt sich, dass eine klare Definition der Anforderungen am

---

<sup>45</sup> Vgl. Seelmann und Hicking (2022), S. 221; vgl. Wieczorrek und Mertens (2011), S. 104 ff.; vgl. Beck et al. (2001), Internetquelle

<sup>46</sup> Jakoby (2024), S. 156

<sup>47</sup> Vgl. Jordan (1997), S. 26; vgl. Goldmann et al. (1995), S. 41; vgl. Jakoby (2024) S. 156

<sup>48</sup> Vgl. Alpar et al. (2023), S. 414 f.

Anfang des Projekts unerlässlich ist, unabhängig vom gewählten Vorgehensmodell.

Auch bei Ausschreibungen behält das Lastenheft weiterhin seine Relevanz.

Während das Lastenheft in diesem Zusammenhang weiterhin eine bedeutende Rolle spielt, entsteht das Pflichtenheft tendenziell implizit und entwickelt sich parallel zur fortschreitenden Softwareentwicklung. Dabei kommen Instrumente wie Backlogs zur Speicherung von Anforderungen, Visions und Goals zur Definition von Anforderungen an das Gesamtsystem, Epics für noch unklare oder sehr grobe Anforderungen sowie User Stories für konkrete funktionale Anforderungen zum Einsatz.<sup>49</sup>

Trotzdem kann es auch hier Sinn ergeben eine grobe Lösungskonzeption zu entwickeln, welche dazu dient, „den Projektstart in die richtige Richtung zu lenken und auf vertraglicher Ebene zwischen Auftraggeber und Auftragnehmer ein Mindestmaß an Verbindlichkeit zu schaffen.“<sup>50</sup>

## **2.4 Definition und Bedeutung von Materialzertifikaten**

Das Unternehmen produziert seine Produkte gemäß den Normen [...] und [...].

Diese Normen fordern eine lückenlose Nachverfolgbarkeit aller extern zugekauften Komponenten durch Materialzertifikate, die in den Produkten verwendet werden.<sup>51</sup>

Materialzertifikate oder Prüfbescheinigungen sind Dokumente, die die Übereinstimmung eines Materials mit den vereinbarten Spezifikationen und Qualitätsanforderungen bestätigen. Sie spielen eine entscheidende Rolle in der Qualitätssicherung und Rückverfolgbarkeit von Materialien, insbesondere in der metallverarbeitenden Industrie.<sup>52</sup>

Die Norm DIN EN 10204:2005-01 beschreibt die verschiedenen Arten von Materialzertifikaten, die für metallische Werkstoffe relevant sind.

Sie definiert verschiedene Typen von Abnahmeprüfzeugnissen und legt fest, welche Informationen und Nachweise in einem solchen Dokument enthalten sein müssen. Obwohl es sich um eine europäische Norm handelt, ist sie weltweit anerkannt. Darüber hinaus ist die

---

<sup>49</sup> Vgl. Hoppen (2015) S. 750

<sup>50</sup> Krypczyk und Bochkor (2018), S. 227

<sup>51</sup> [...]

<sup>52</sup> Vgl. Vorest AG (o.D.), Internetquelle; vgl. LB MetallService AG (o.D.), Internetquelle

Anwendung dieser Norm nicht auf metallische Produkte beschränkt, sondern kann auch auf nichtmetallische Produkte angewendet werden.<sup>53</sup>

Für das beschriebene Unternehmen sind vor allem Prüfbescheinigungen vom Typ 3.1 (Abnahmeprüfzeugnis) von Bedeutung. Diese Zertifikate stellen sicher, dass die Materialeigenschaften anhand spezifischer Prüfungen nachgewiesen wurden und bestätigen, dass die gelieferten Materialien den festgelegten Qualitäts- und Kundenanforderungen entsprechen. Wichtig ist, dass bei der Prüfbescheinigung vom Typ 3.1 die Prüfung durch einen Prüfer des Herstellers erfolgt, der jedoch nicht der Produktionsabteilung angehören darf.<sup>54</sup>

### **3. Hauptteil**

Im Hauptteil dieser Arbeit wird zunächst das Lastenheft entwickelt, gefolgt von der Erstellung des Pflichtenhefts. Der Aufbau beider Dokumente orientiert sich an der in Kapitel 2.2 vorgestellten Vorgehensweisen, aus der verschiedene Aspekte ausgewählt wurden.

#### **3.1 Konzeption des Lastenheftes**

##### **3.1.1 Das Unternehmen**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

##### **3.1.2 Ausgangssituation**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

##### **3.1.3 Ziele des Projekts**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

##### **3.1.4 Anforderungen**

###### **3.1.4.1 Funktionale Anforderungen**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

---

<sup>53</sup> Vgl. Lohse et al (2016), S. 21; vgl. Friederich (2006), S. 48 ff., vgl. Vorest AG (o.D.), Internetquelle

<sup>54</sup> Vgl. DIN EN 10204:2005-01 (2005), o.S.; vgl. Hinsch et al. (2022), S. 172

### **3.1.4.2 Nicht-funktionale Anforderungen**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

### **3.1.5 Bestehende Architektur und Definition von Schnittstellen**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

## **3.2 Konzeption des Pflichtenheftes**

### **3.2.1 Soll-Zustand**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

### **3.2.2 Systemarchitektur**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

### **3.2.3 Benutzeroberflächen**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

### **3.2.4 Grobe Projektplanung**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

### **3.2.5 Weiterentwicklung und mögliche Anpassungen**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

## **4. Schlussbetrachtung**

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

Als Erstes wurde das Lastenheft konzipiert. Hier wurde zunächst die Ist-Situation erfasst und die damit verbundenen Probleme, wie [...] und [...], beschrieben. Anschließend wurden Ziele wie [...] und [...], die [...] entstehen, definiert. Im Folgenden wurden die funktionalen Anforderungen an das System festgelegt sowie die nicht-funktionalen Anforderungen zu den Themen Sicherheit, Belastbarkeit und Benutzbarkeit aufgelistet.

Im Anschluss wurde das Pflichtenheft entwickelt, welches konkrete technische Lösungen für die im Lastenheft beschriebenen Anforderungen definiert.

Im Mittelpunkt standen dabei der angestrebte Soll-Zustand, die Systemarchitektur, die Benutzeroberflächen, der grobe Projektplan sowie mögliche Weiterentwicklungen und Anpassungen des Systems.

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

In Anbetracht der Tatsache, dass *[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*, kann das beschriebene Vorgehen zur Erstellung eines Lasten- und Pflichtenheftes kritisch hinterfragt werden. Auf diese Problematik wurde bereits in Kapitel 2.3.3 ausführlich eingegangen. Da *[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*, die auf *[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]* zurückzuführen waren, erscheint die Erstellung eines Lasten- und Pflichtenheftes vor diesem Hintergrund gerade für dieses Projekt sinnvoll. Ob das beschriebene Vorgehen tatsächlich erfolgreicher ist, lässt sich erst nach der vollständigen Implementierung und dem Abschluss des Projekts bewerten.

Für diese Konzeption wurde ein selbst entwickelter Aufbau gewählt, der Teile der vorgestellten Herangehensweisen (Kapitel 2.2) enthält. Bei größeren Softwareprojekten empfiehlt es sich, die VDI-Norm 3694 stärker als Orientierung zu nutzen, da diese eine umfassendere Struktur mit zusätzlichen Unterkapiteln bietet. Die Norm beinhaltet Aspekte wie die detaillierte Projektplanung sowie die Definition von Anforderungen an die Systementwicklung, -inbetriebnahme und -einsatz, die in dieser Arbeit aufgrund des begrenzten Umfangs nicht behandelt werden konnten. Auch auf formale Aspekte des Lasten- und Pflichtenheftes wie

einem Deckblatt, einer Version und einem Inhaltsverzeichnis musste aus diesem Grund verzichtet werden.

Für die Definition der Materialzertifikate musste teilweise auf graue Literatur zurückgegriffen werden.

*[Der Abschnitt wurde aus der öffentlichen Version dieser Arbeit entfernt, da er vertrauliche Daten enthält.]*

Das Thema Vorgehensmodelle in der Softwareentwicklung konnte nur angerissen werden. In der Praxis werden häufig auch hybride Modelle eingesetzt, die sich nicht eindeutig einem einzelnen Modell zuordnen lassen.<sup>55</sup>

---

<sup>55</sup> Vgl. Broy und Kuhrmann (2021), S. 86

#### **IV. Literaturverzeichnis - Buchquellen**

- Alpar, P./Alt, R./Bensberg, F. (u.a.) (2023): Anwendungsorientierte Wirtschaftsinformatik - Strategische Planung, Entwicklung und Nutzung von Informationssystemen, 10. Aufl., Wiesbaden: Springer Vieweg
- Bär, C./Fiege, J./Weiß, M. (2017): Anwendungsbezogenes Projektmanagement - Praxis und Theorie für Projektleiter, Berlin: Springer Vieweg
- Boehm, B. (1981): Software Engineering Economics, New Jersey: Prentice Hall
- Brandt-Pook, H./Kollmeier, R. (2020): Softwareentwicklung kompakt und verständlich - Wie Softwaresysteme entstehen, 3., verbesserte Aufl., Wiesbaden: Springer Vieweg
- Broy, M./Kuhmann, M. (2013): Projektorganisation und Management im Software Engineering, Berlin: Springer Vieweg
- Broy, M./Kuhmann, M. (2021): Einführung in die Softwaretechnik. Berlin: Springer Vieweg
- DIN EN 10204:2005-01 (2005): Metallische Erzeugnisse - Arten von Prüfbescheinigungen, Berlin: Beuth Verlag
- DIN 69901-5 (2009): Projektmanagement - Projektmanagementsysteme - Teil 5: Begriffe, Berlin: Beuth Verlag
- Duschl, D. (2024): Software-Fehler erkennen und vermeiden, Wiesbaden: Springer Vieweg
- Eisele, O./Harlacher, M./Lennings, F. (2023): Bedarfsgerechte Auswahl und Einführung von KI-Anwendungen, in: Stowasser, S. (Hrsg.): Künstliche Intelligenz (KI) und Arbeit - Leitfaden zur soziotechnischen Gestaltung von KI-Systemen, Berlin: Springer Vieweg
- Ernst, H./Schmidt, J./Beneken, G. (2023): Grundkurs Informatik - Grundlagen und Konzepte für die erfolgreiche IT-Praxis – Eine umfassende Einführung, 8. Aufl., Wiesbaden: Springer Vieweg
- Friederich, I. (2006): Nach einem Jahr mit der neuen Norm für Prüfbescheinigungen- Viele Fragen offen, in: QZ – Qualität und Zuverlässigkeit, 2006 (5), München: Hanser, S. 48-51

- Gagliardi, V. (2021): Decoupled Django - Understand and Build Decoupled Django Architectures for JavaScript Front-ends, New York: Apress
- Goldman, S./Nagel, R./Preiss, K. (1995): Agile Competitors and Virtual Organizations - Strategies for Enriching the Customer, New York: Van Nostrand Reinhold
- Goll, J. (2011): Methoden und Architekturen der Softwareentwicklung. Wiesbaden: Vieweg+Teubner Verlag
- Groß, C./Pfenning, R. (2019): Digitalisierung in Industrie, Handel und Logistik Leitfaden von der Prozessanalyse bis zur Einsatzoptimierung, 2. Aufl., Wiesbaden: Springer Gabler
- Hahn, M. (2020): Webdesign – Das Handbuch zur Webgestaltung, 3. Aufl., Bonn: Rheinwerk Verlag
- Hammerschall, U./Beneken, G. (2013): Software Requirements, München: Pearson Deutschland GmbH
- Hinsch, M./Klarner, A./Berlinger, C. (2022): Einführung in die DEMAR - Strategien und Umsetzung des Zulassungswesens für Luftfahrzeuge der Bundeswehr, Berlin: Springer Vieweg
- Hoppen, P. (2015): Software-Anforderungsdokumentation, in: Bartsch, M./Grützmacher, M./Härtling, N. (u.a.): Computer und Recht, 2015 (11), Köln: Otto Schmidt, S. 747-760
- ISO/IEC/IEEE (Hrsg.) (2018): 29148:2018-11 - Systems and software engineering – Life cycle processes – Requirements engineering, New York City: IEEE
- Jakoby, P. (2024): Agilität in der digitalen Transformation: Scrum als Schlüssel zum Erfolg? in: Bruns, P./Bruns, W. (Hrsg.): Praxishandbuch Kompetenzen in der Digitalen Transformation der Arbeit, Wiesbaden: Springer VS, S. 155-190
- Jordan, J. (1997): Enablers for Agile Virtual Enterprise Integration, in: Agility & Global Competition, 1 (3), New Jersey: Wiley, S. 26-46
- Kaufmann, J./Mülder, W. (2023): Grundkurs Wirtschaftsinformatik - Eine kompakte und praxisorientierte Einführung, 10. Aufl., Wiesbaden: Springer Vieweg

- Krypczyk, V./Bochkor, O. (2018): Handbuch für Softwareentwickler, Bonn: Rheinwerk Verlag
- Lettau, C. (2000): Das Web-Pflichtenheft, Bonn: MITP-Verlag GmbH
- Lohse, W./Laumann, J./Wolf, C. (2016): Stahlbau 1 - Bemessung von Stahlbauten nach Eurocode mit zahlreichen Beispielen, 25. Aufl., Wiesbaden: Springer Vieweg
- Madauss, B.-J. (2020): Projektmanagement - Theorie und Praxis aus einer Hand, 8. Aufl., Berlin: Springer Vieweg
- Metzner, A. (2020): Software Engineering – Kompakt, München: Carl Hanser Verlag München
- Osterhage, W. (2016): Qualität in der Softwareentwicklung, in: Leyh, C./Schäffer, T. (Hrsg.): Digitale Kompetenzen (HMD), 61 (1), Wiesbaden: Springer Vieweg, S. 153-168
- Royce, W. (1970): Managing the development of large systems, in: Proceedings of IEEE Wescon (26), Los Angeles: TRW, S. 1-9
- Seilmann, V./Hicking, J. (2022): Projektmanagement, in: Schuh, G./Zeller, V./Stich, V. (Hrsg.): Digitalisierungs- und Informationsmanagement - Handbuch Produktion und Management 9, Berlin: Springer Vieweg, S. 213-248
- Stender, P. (2011): Webprojekte realisieren nach neuesten OOP-Kriterien – Ein Workshop über die Kooperation von PHP/XSL/JavaScript, Wiesbaden: Vieweg+Teubner Verlag
- Teich, I./Kolbenschlag, W./Reiners, W. (2008): Der richtige Weg zur Softwareauswahl – Lastenheft, Pflichtenheft, Compliance, Erfolgskontrolle, Berlin: Springer
- VDI (2001): VDI 2519 Blatt 1 - Vorgehensweise bei der Erstellung von Lasten-/Pflichtenheften, Berlin: Beuth Verlag
- VDI (2014): VDI 3694 - Lastenheft/Pflichtenheft für den Einsatz von Automatisierungssystemen, Berlin: Beuth Verlag
- Wieczorrek, H. W./Mertens, P. (2011): Management von IT-Projekten - Von der Planung zur Realisierung, 4. Aufl., Heidelberg: Springer Berlin

## **V. Literaturverzeichnis - Internetquellen**

Beck, K./Beedle, M./Beenekum A.v. (u.a.) (2001): Manifest für Agile Softwareentwicklung. agilemanifesto, <https://agilemanifesto.org/iso/de/manifesto.html> (Abgerufen am 20.10.2024)

LB MetallService AG (o.D.): Materialprüfbescheinigungen. LB MetallService AG, <https://www.lb-metallservice.ch/de/materialpruefzertifikate/> (Abgerufen am 20.10.2024)

Vorest AG (o.D.): Prüfbescheinigungen & deren Arten nach DIN EN 10204. Vorest AG, <https://www.vorest-ag.com/Qualitaetssicherung-Methoden/Wissen/pruefbescheinigungen> (Abgerufen am 20.10.2024)