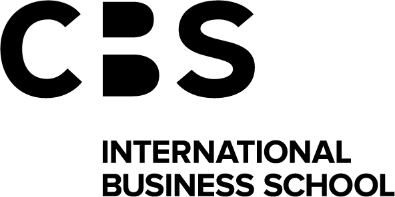
****

**CBS  
INTERNATIONAL BUSINESS SCHOOL**

**KIOSKSYSTEME ZUR KUNDENORIENTIERUNG IM STATIONÄREN HANDEL**

**Bachelorarbeit**

vorgelegt in teilweiser Erfüllung der Voraussetzungen für die Erlangung des Grades eines

**BACHELOR OF SCIENCE (B. Sc.)**

im Programm Wirtschaftsinformatik  
mit Spezialisierung in Software-Entwicklung und Systeminfrastrukturen

Daniel Gilbers  
Immatrikulationsnummer: 2201318

Betreuer: Prof. Dr. Steffen Stock

Erftstadt, 27. November 2024

**Inhaltsverzeichnis**

Abbildungsverzeichnis III

Tabellenverzeichnis IV

Abkürzungsverzeichnis V

1 Einleitung (1 Seite) 1

2 Kundenorientierung im stationären Handel (3 Seiten) 2

3 Kiosksysteme (5 Seiten) 3

3.1 Anforderungen an Kiosksystemsoftware 3

3.2 Vorgehensmodell zur Softwareentwicklung 4

4 Konzept zu Kiosksystemen zur Kundenorientierung im stationären Handel (12 Seiten) 7

5 Umsetzung eines Kiosksystems zur Kundenorientierung im stationären Handel (8 Seiten) 8

6 Fazit (2 Seiten) 9

Literaturverzeichnis 10

Anhang 12

KI-Tools & KI-Nutzung 13

# Abbildungsverzeichnis

Abbildung 1: Vorgehensmodell Extreme Programming (XP). 5

# Tabellenverzeichnis

Tabelle 1: Product quality model nach ISO/IEC 25010. 4

# Abkürzungsverzeichnis

TDD Test Driven Development

WCAG Web Content Accessibility Guidelines

XP Extreme Programming

# Einleitung (1 Seite)

Erläuterung der Motivation (allgemeines Problem, Unternehmen und unternehmensspezifisches Problem), der Forschungsfrage und des Aufbaus der Arbeit (sowie des Forschungsdesigns).

Ziel ist die Untersuchung wie ein Kiosksystem zur Kundenorientierung im stationären Handel gestaltet sein sollte. Dazu müssen folgende Unterfragen geklärt werden:

1. Wie orientieren sich Kunden im stationären Handel?
2. Wie können Kiosksysteme dabei unterstützen?
3. Wie sollte die Software für ein solches Kiosksystem gestaltet sein?

# Kundenorientierung im stationären Handel (3 Seiten)

Definition

# Kiosksysteme (5 Seiten)

Ein Kiosksystem ist ein interaktives, computergestütztes System, das an öffentlich zugänglichen Orten aufgestellt wird, um Benutzern Informationen oder Transaktions-möglichkeiten zur Verfügung zu stellen. Einsatzorte können z. B. Bahnhöfe, Flughäfen, Einzelhandelsgeschäfte oder Bankfilialen sein. Sie bieten einen eingeschränkten Zugang zu vorgewählten Anwendungsprogrammen. Sofern Webbrowser zu den Anwendungs-programmen gehören, sind diese ebenfalls in ihrem Nutzungsumfang eingeschränkt. Zur Interaktion werden in der Regel Touchscreens verwendet. Da ihr Einsatz auf einen Dauerbetrieb ohne betreuendes Personal ausgelegt ist, sind Kiosksystems robust gegenüber Fehlbedienungen oder Manipulationsversuchen.[[1]](#footnote-2)

Kiosksysteme im Handel werden zu unterschiedlichen Zwecken eingesetzt, die sich aus ihren konkreten Funktionen ableiten lassen. Dazu zählen in erster Linie eine personalunabhängige Informations-, Präsentations- und Beratungsfunktion. Des Weiteren zählen die Erlebnisorientierung beim Einkauf und die Umsatzgenerierung zu den häufigsten Einsatzgebieten. Dafür bietet ein Kiosksystem z. B. die Möglichkeit als Plattform für Retail Media zu dienen. Dies ist Werbung für Marken oder Produkte direkt im Ökosystem des Retailers und damit dort, wo ihre Wahrnehmbarkeit und Akzeptanz hoch ist.[[2]](#footnote-3) Darüber hinaus bieten Kiosksysteme aber auch die Möglichkeit, Servicefunktionen wie ein Beschwerdemanagement anzubieten und das Unternehmensimage im Sinne einer besseren Serviceorientierung zu verbessern. Ein Einsatz, egal mit welchem Fokus, führt zu einer Verbesserung des Kundenerlebnis und ihrer Loyalität und sollte daher von Unternehmen in Betracht gezogen werden.[[3]](#footnote-4)

Da die Einsatzmöglichkeiten eines Kiosksystems maßgeblich von der eingesetzten Software bestimmt werden, wird in den folgenden Unterkapiteln näher auf diese eingegangen. Unterkapitel 3.1 beschäftigt sich mit den funktionalen und nicht-funktionalen Anforderungen an die Kiosksystemsoftware. In Unterkapitel 3.2 wird dann das in dieser Arbeit verwendete Vorgehensmodell zur Softwareentwicklung beschrieben.

## Anforderungen an Kiosksystemsoftware

Als Basis der Anforderungen für Kiosksystemsoftware dienen die neun in Tabelle 1 aufgeführten Qualitätsmerkmale der internationalen Norm für System- und Softwarequalität (ISO/IEC 25010). Diese beschreiben ein funktionales und acht nicht-funktionale Merkmale der Softwarequalität, aus denen sich konkrete Anforderungen ableiten lassen. Die Functional suitability deckt als das funktionale Merkmal die Kernaufgaben der Anwendung ab. Jedes Merkmal besteht aus mehreren Untermerkmalen und wird je nach Art der Anwendung unterschiedlich priorisiert.[[4]](#footnote-5)

|  |  |  |
| --- | --- | --- |
| **Product Quality** | | |
| Functional suitability | Interaction capability | Maintainability |
| Performance efficiency | Reliability | Flexibility |
| Compatibility | Security | Safety |

Tabelle 1: Product quality model nach ISO/IEC 25010.[[5]](#footnote-6)

Die Merkmale Interaction capability, Performance efficiency und Reliability haben eine hohe Priorität für Kiosksysteme. Die Interaction capability beschreibt die Fähigkeit zur Interaktion zwischen Benutzer und Anwendung, die bei Kiosksystemen möglichst einfach und ohne vorherige EDV-Erfahrung möglich sein sollte. Da ältere Menschen und Menschen mit Behinderungen häufig Probleme bei der Nutzung von Kiosksystemen haben, sollte ein Barrierefreier Zugang möglich sein.[[6]](#footnote-7) Um dies umzusetzen, können die Web Content Accessibility Guidelines (WCAG) als internationaler Standard genutzt werden.[[7]](#footnote-8) Die Performance efficiency beschreibt die Fähigkeit der Anwendung ihre Funktionen in vorgegebener Zeit und mit vorgegebenem Ressourcenverbrauch auszuführen. Dafür muss z. B. die Reaktionsfähigkeit berücksichtigt werden, da Benutzer nur ein bis zwei Sekunden auf abgerufene Informationen zu warten bereit sind.[[8]](#footnote-9) Das dritte Merkmal, Reliability, bezieht sich auf die Fähigkeit der Anwendung fehlerfrei und damit zuverlässig zu funktionieren. Da Kiosksysteme personalunabhängig eingesetzt werden sollen, ist die Zuverlässigkeit des Systems eine Grundvoraussetzung.

Die restlichen Merkmale müssen je nach Art der Anwendung priorisiert werden. Das Merkmal Maintainability bezieht sich auf den Grad der Modifizierbarkeit. Dazu zählen sowohl die Wartbarkeit als auch die Erweiterbarkeit der Anwendung. Die Flexibility bezieht sich auf die Anpassungsfähigkeit an unterschiedliche Umgebungen. Dies beinhaltet z. B. die Fähigkeiten, auf unterschiedlicher Hardware oder trotz Verlust der Internetverbindung zu funktionieren. Compatibility beschreibt die Fähigkeit Daten mit anderen Anwendungen, wie z. B. einem Enterprise-Resource-Planning-System, auszutauschen. Security beschreibt die Fähigkeiten der Anwendung, Daten nur mit entsprechender Berechtigung zugänglich zu machen. Dies ist z. B. notwendig, wenn über das Kiosksystem auf personenbezogene Daten zugegriffen werden soll. Datenschutzaspekte, wie z. B. die Umsetzung der Verordnung (EU) 2016/679 (Datenschutz-Grundverordnung), fallen ebenfalls in diesen Bereich. Das letzte Merkmal, Safety, bezieht sich auf die Fähigkeit, den Benutzer vor Schaden zu bewahren.

## Vorgehensmodell zur Softwareentwicklung

Als Vorgehensmodell für die Softwareentwicklung wird in dieser Arbeit eine Variante des Extreme Programming (XP) gewählt. XP bietet einen agilen Ansatz, der einen flexiblen Umfang des Endprodukts erlaubt und für den Entwickler leicht modifizierbar ist. Durch einwöchige Iterationen können Funktionalitäten schnell implementiert werden, ohne dass ein großer Dokumentationsaufwand entsteht. Darüber hinaus werden durch Test Driven Development (TDD) bereits während der Entwicklung Tests erstellt, die eine hohe Softwarequalität ermöglichen. Das Vorgehen und die zu erstellenden Artefakte sind in Abbildung 1 dargestellt.[[9]](#footnote-10)

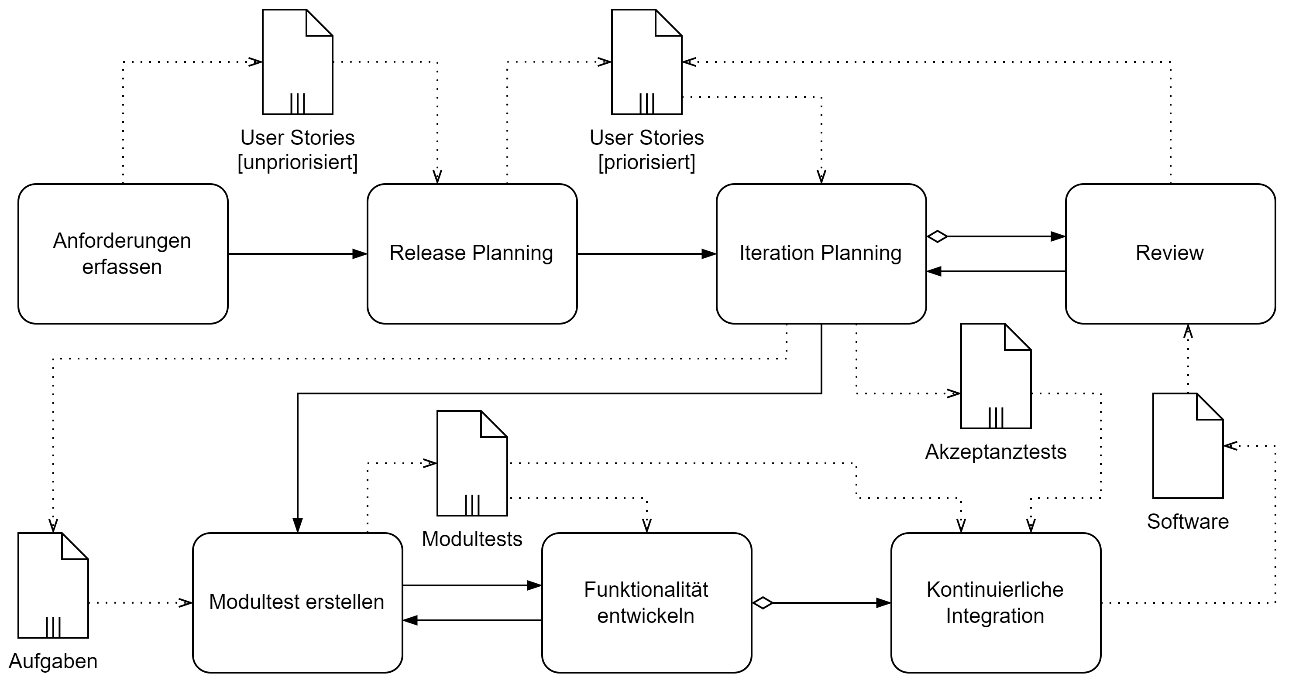


Abbildung 1: Vorgehensmodell Extreme Programming (XP).[[10]](#footnote-11)

Zu Beginn werden die Anforderungen aufgenommen und in User Stories festgehalten. Eine User Story wird dabei vom Kunden formuliert und beschreibt in wenigen Sätzen eine gewünschte Funktionalität. Dabei ist es noch nicht notwendig auf technische Details einzugehen. Nachdem die User Stories erstellt wurden, werden sie im Rahmen des Release Planning priorisiert. Dabei wird entschieden, welche User Stories in einer ersten Version der Software umgesetzt werden sollen und wie diese priorisiert werden. Darüber hinaus wird der jeweilige Arbeitsumfang geschätzt.[[11]](#footnote-12)

Die wichtigsten User Stories werden in der ersten Iteration umgesetzt. Die Anzahl richtet sich nach dem geschätzten Arbeitsumfang der User Stories, da eine Iteration nur eine Woche dauert. Im Iteration Planning wird für jede gewählte User Story ein Akzeptanztest erstellt. Diese Tests beschreiben konkrete Anwendungsfälle der jeweiligen User Story und dienen der Überprüfung, ob eine User Story erfolgreich umgesetzt wurde. Zusätzlich werden die User Stories von den Entwicklern in kleinere Aufgaben aufgeteilt. Dies sind detailliertere und technischere Anforderungen, die zur Erfüllung der User Story umgesetzt werden müssen. Falls bereits eine Iteration abgeschlossen wurde, kann die Project Velocity gemessen werden. Die Project Velocity ist gleich der Summe des geschätzten Arbeitsumfangs der pro Iteration fertiggestellten User Storys. Durch sie kann genauer geplant werden, wie viele User Storys pro Iteration bearbeitet werden können.[[12]](#footnote-13)

Während einer Iteration werden die Aufgaben einzeln abgearbeitet, wobei nach dem Prinzip des TDD vorgegangen wird. Das bedeutet, dass immer zuerst ein passender Modultest erstellt wird. Sobald der Modultest vorliegt, wird mit der eigentlichen Entwicklung begonnen. Ziel der Entwicklung ist es, den Modultest zu bestehen. Dabei sollen nur die Funktionalitäten implementiert werden, die zum Bestehen des Modultests notwendig sind. Funktionalitäten, die vielleicht in Zukunft benötigt werden, sollen bewusst nicht implementiert werden, da dies die Entwicklung verlangsamen würde. Ist der Modultest bestanden, kann er zusammen mit dem Programmcode über Kontinuierliche Integration ausgeliefert werden. In diesem Schritt werden alle bis dahin erstellten Akzeptanz- und Modultests als Regressionstests durchgeführt, um sicherzustellen, dass durch eine neue Funktionalität keine neuen Fehler in bereits bestehenden Teilen der Software auftreten. Falls die Regressionstests bestanden wurden, wird die neu entwickelte Funktionalität ausgeliefert.[[13]](#footnote-14)

Am Ende einer Iteration wird dem Kunden in einem Review der aktuelle Stand der Software präsentiert. Das Feedback oder geänderte Anforderungen können dabei in neue User Stories einfließen. Auch die Priorisierung kann sich ändern und in die nächste Iteration einfließen.[[14]](#footnote-15)

Um bei XP sicherzustellen, dass der Code der Software lesbar, verständlich, wartbar und erweiterbar bleibt, muss kontinuierliches Refactoring betrieben werden. Dabei wird der Code umstrukturiert, ohne die Funktionalität zu verändern. Im Unterschied zum klassischen XP wird im Rahmen dieser Arbeit mangels Entwicklerteam allerdings auf Paarprogrammierung verzichtet. Darüber hinaus werden auf Grund der begrenzten Bearbeitungszeit dieser Arbeit wahrscheinlich nur zwei bis drei Iterationen durchgeführt.[[15]](#footnote-16)

# Konzept zu Kiosksystemen zur Kundenorientierung im stationären Handel (12 Seiten)

Systemanalyse (4 Seiten)

User Storys (Anforderungen), Marktsichtung für Fertigprodukte (SW-Module).

Systementwurf (8 Seiten)

Aufgaben (Anforderungen detailliert), SW-Architektur, Datenbankentwurf, Styleguide/Wireframe.

# Umsetzung eines Kiosksystems zur Kundenorientierung im stationären Handel (8 Seiten)

Exemplarische Umsetzung eines Prototyps anhand des Konzepts. Kritische Auseinandersetzung mit Umsetzung.

# Fazit (2 Seiten)

Fazit und kritische Auseinandersetzung zu Konzept (ggf. Ausblick auf weiteren Forschungsbedarf).

# Literaturverzeichnis

**Beck / Andres 2004**

Beck, Kent; Andres, Cynthia: Extreme Programming Explained. Embrace Change. 2. Aufl. Boston 2004.

**BVDW e. V. 2024**

Bundesverband Digitale Wirtschaft (BVDW) e. V.: RMC (Retail Media Circle) – Bundesverband Digitale Wirtschaft (BVDW) e. V. Https://www.bvdw.org/gremien/retail-media-circle/, 2024, Abruf am 9. Oktober 2024.

**Fischer 2002**

Fischer, Lars: Kiosksysteme im Handel. Einsatz, Akzeptanz und Wirkung. Wiesbaden 2002.

**ISO/IEC 2023**

ISO/IEC: ISO/IEC 25010:2023(E). Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – Product quality model. 2. Aufl. Vernier, Genf 2023.

**Johnson 2021**

Johnson, Jeff: Designing with the Mind in Mind. Simple Guide to Understanding User Interface Design Guidelines. 3. Aufl. Cambridge 2021.

**Lackes et al. 2018**

Lackes, Richard; Siepermann, Markus; Kollmann, Tobias: Kiosksystem • Definition | Gabler Wirtschaftslexikon. Https://wirtschaftslexikon.gabler.de/definition/kiosksystem-38106/version-261532, 2018, Abruf am 5. Oktober 2024.

**Lee et al. 2023**

Lee, Yuryeon; Park, Sunyuong; Park, Jaehyun; Kim, Hyun: Comparative Analysis of Usability and Accessibility of Kiosks for People with Disabilities. In: Applied Sciences 13 (2023) 5, Nr. 3058.

**Nah 2003**

Nah, Fiona: A Study on Tolerable Waiting Time: How Long Are Web Users Willing to Wait? In: AMCIS 2003 Proceedings (2003), S. 2212 – 2222.

**Sachani 2023**

Sachani, Dipakkumar: The Role of Kiosks in Omni-Channel Retail Strategies: A Market Perspective. In: Journal of Computing and Digital Technologies 1 (2023) 1, S. 62 – 75.

**Stone et al. 2005**

Stone, Debbie; Jarrett, Caroline; Woodroffe, Mark; Minocha, Shailey: User Interface Design and Evaluation. San Francisco 2005.

**Wells 2013**

Wells, Don: Extreme Programming Rules. Http://www.extremeprogramming.org/rules.html, 2013, Abruf am 5. Oktober 2023.

**W3C 2023**

World Wide Web Consortium: WCAG 2 Overview | Web Accessibility Initiative (WAI) | W3C. Https://www.w3.org/WAI/standards-guidelines/wcag/, 2023, Abruf am 5. Oktober 2024.

# Anhang

Anhang

# KI-Tools & KI-Nutzung

|  |  |  |
| --- | --- | --- |
| Kapitel | KI-Tools | Beschreibung der Verwendung |
|  |  |  |
|  |  |  |

**Erklärung an Eides statt**

Hiermit versichere ich, dass ich diese Arbeit selbständig und ohne fremde Hilfe verfasst habe. Dabei habe ich keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Ich habe dabei keine urheberrechtlich geschützten Werke oder Werkteile unverändert übernommen oder in einer Weise umgearbeitet übernommen. Die Stellen in der Arbeit, die dem Wortlaut oder dem Sinn nach anderen Werken und Quellen – einschließlich der Quellen aus dem Internet – entnommen sind, sind von mir unter der Angabe der Quelle als Zitat kenntlich gemacht.

**Einräumung von Nutzungsrechten**

Zur Überprüfung der Arbeit auf Verstöße gegen das Urheberrecht und Plagiate setzt die CBS neben einer manuellen Prüfung auch sogenannte web-basierte Anti-Plagiatssoftware ein. Zur Durchführung der Überprüfung meiner Arbeit räume ich der CBS und ihren externen Dienstanbietern das Recht ein, die Arbeit auf elektronischem Weg zu vervielfältigen, zu speichern und zwischenzuspeichern sowie zeitlich unbeschränkt für Vergleichszwecke bei anderen Prüfungsarbeiten heranzuziehen. Ich willige dahingehend ein, dass meine Arbeit im Rahmen der Plagiatsprüfung gespeichert und genutzt wird, insbesondere an Anbieter einer web-basierten Plagiatssoftware auch im Ausland übermittelt werden kann, die diese nur für diesen Zweck verarbeitet und nutzt.

**Nutzung von KI-gestützten Tools**

Ich erkläre hiermit, dass ich beim Einsatz von generativen KI-gestützten Tools für die vorliegende Arbeit diese Werkzeuge in dem Verzeichnis `KI-Tools & KI-Nutzung` aufgeführt habe. Dort ist präzisiert, inwieweit sie zum Zwecke der Strukturierung, Formulierung, inhaltlichen Recherche, Programmierhilfe, Übersetzung, etc. verwendet worden sind.

Bei der Erstellung dieser Arbeit habe ich durchgehend eigenständig und beim Einsatz generativer KI-gestützter Schreibwerkzeuge stets steuernd gearbeitet. Jede Quelle, die KI-basiert vorgeschlagen wurde, habe ich überprüft, kritisch im Kontext reflektiert und entsprechend der Zitierschreibweise gekennzeichnet. Ich bin mir bewusst, dass ich, falls ich generative KI-basierte Tools zur Erstellung dieser Arbeit verwendet habe, für durch die KI-Tools generierte falsche oder verzerrte Inhalte, falsche Referenzen, Verstöße gegen Datenschutz- und Urheberrechtsgesetze sowie die Erzeugung eines Plagiats verantwortlich bin.

**Datenschutzerklärung**

In einer Prüfungsarbeit können auch Aussagen über persönliche und sachliche Verhältnisse des betreffenden Studierenden oder anderer Personen enthalten sein. Die Erhebung, Speicherung und Nutzung solcher Daten sind nur bei Einwilligung des Betroffenen möglich. In diesem Zusammenhang versichere ich, dass alle betroffenen Personen (z.B. gegebenenfalls Interviewpartner) einer Veröffentlichung zugestimmt haben und ich geklärt habe, ob eine Anonymisierung gewünscht ist sowie dass alle personenbezogenen Daten derjenigen Personen anonymisiert wurden, die einer Veröffentlichung nur in anonymisierter Form zugestimmt haben.

Erftstadt, 27.11.2024

Daniel Gilbers, BR BSc WI 22W D

1. Vgl. Fischer 2002, S. 5 f. sowie Lackes et al. 2018. [↑](#footnote-ref-2)
2. Vgl. BVDW 2024. [↑](#footnote-ref-3)
3. Vgl. Fischer 2002, S. 159 f. und Sachani 2023, S. 71 f. [↑](#footnote-ref-4)
4. Vgl. hierzu und zu Folgenden ISO/IEC 2023, S. 2 ff. [↑](#footnote-ref-5)
5. In Anlehnung an: ISO/IEC 2023, S. 10. [↑](#footnote-ref-6)
6. Vgl. Lee et al. 2023, S. 1 ff. [↑](#footnote-ref-7)
7. Vgl. W3C 2023. [↑](#footnote-ref-8)
8. Vgl. Nah 2003, S. 2220 und Johnson 2021, S. 244 ff. [↑](#footnote-ref-9)
9. Vgl. Beck / Andres 2004, S. 2 ff. sowie hierzu und zu Folgenden Wells 2018. [↑](#footnote-ref-10)
10. In Anlehnung an: Wells 2018. [↑](#footnote-ref-11)
11. Vgl. Beck / Andres 2004, S. 44 f., 47 und 91 ff. [↑](#footnote-ref-12)
12. Vgl. Beck / Andres 2004, S. 46. [↑](#footnote-ref-13)
13. Vgl. Beck / Andres 2004, S. 49 f. [↑](#footnote-ref-14)
14. Vgl. Beck / Andres 2004, S. 61 und 91 ff. [↑](#footnote-ref-15)
15. Vgl. Beck / Andres 2004, S. 42 f. und 51 ff. [↑](#footnote-ref-16)