



Konzeption eines Systems zur dreidimensionalen Visualisierung der Auslastung von Rechenzentren an den Beispielen der Bundesagentur für Arbeit und eines Automobilherstellers

Masterarbeit

zur Erlangung des Titels Master of Science im Fach
Wirtschaftsinformatik und Electronic Government

vorgelegt von

Lukas Sprinck

Angefertigt am
Lehrstuhl für Wirtschaftsinformatik (Prozesse und Systeme)
der Universität Potsdam
1. Gutachter: Dr. Eldar Sultanow
2. Gutachter: Dr. André Ullrich

Potsdam, den 09.08.2019

Abstract

Mit fortschreitender Digitalisierung wächst die Menge an verfügbaren Daten, die häufig essenzieller Bestandteil der Wertschöpfung sind [Digi2018]. Für die Verarbeitung der Datenmengen werden hochverfügbare Rechenzentren benötigt. Um diese Verfügbarkeit zu gewährleisten, müssen die Daten der Rechenzentren durch sogenannte IT-Operations Analytics Systeme überwacht werden [Rüdi2019]. Bei den meisten Systemen müssen die Benutzer bei der Serverwartung zwischen verschiedenen Tools und innerhalb dieser zwischen mehreren Ansichten wechseln. Nur wenige Tools bieten genügend Flexibilität, die Oberfläche nach eigenen Wünschen zu verändern. Für diese Tools existiert jedoch kein bekanntes Konzept, wie eine Oberfläche zur Überwachung von Rechenzentren gestaltet werden kann.

Ziel ist es, im Rahmen dieser Arbeit ein entsprechendes Konzept zu erstellen. Im Zentrum der konzipierten Oberfläche steht ein dreidimensionales Diagramm, weil dieses es ermöglicht, die gesamte Serverlandschaft eines Unternehmens auf einen Blick darzustellen. Dazu wird folgenden Fragen nachgegangen: Welche Tools existieren im Umfeld von IT-Operations Analytics, inwiefern sind diese Tools gegenüber heutigen Anforderungen unzureichend und wie lässt sich diese Lücke aufheben?

Zur Beantwortung der Fragen werden durch Interviews und Brainstorming mit Benutzern der Bundesagentur für Arbeit und dem Automobilhersteller zunächst Anforderungen gesammelt und Mockups erstellt. Aus diesen entsteht mittels Prototyping ein klickbarer Prototyp, der mit den Benutzern evaluiert wird. Abschließend werden Hinweise für die technische Umsetzung gegeben, die allerdings nicht Teil der vorliegenden Arbeit ist.

Inhaltsverzeichnis

Abkürzungsverzeichnis.....	V
Abbildungsverzeichnis	VI
Tabellenverzeichnis.....	VIII
1 Einleitung.....	9
1.1 Problemstellung.....	9
1.2 Zielstellung.....	10
1.3 Methodisches Vorgehen.....	10
1.4 Gliederung der Arbeit	13
1.5 Anmerkungen zur Schreibweise.....	13
2 Verwendete Methoden und Werkzeuge	14
2.1 Vorgehensmodell.....	14
2.2 Literaturrecherche.....	15
2.3 Experteninterviews	16
2.4 Brainstorming.....	17
2.5 Prototyping	17
2.6 Visualisierungstool für die Mockups.....	18
3 Grundlagen.....	19
3.1 Begriffsdefinitionen – allgemein.....	19
3.1.1 Verbundstrukturen im Rechenzentrum	19
3.1.2 Serverauslastung	20
3.1.3 Wireframe, Mockup und Prototyp	20
3.1.4 IT-Operations Analytics	21
3.1.5 Maschinelles Lernen	21
3.2 GUI und Steuerelemente	22
3.2.1 Grundsätze der Dialoggestaltung	23
3.2.2 Dashboard.....	23
3.2.3 Seitenleiste	24
3.2.4 Tooltip	24
3.2.5 Buttons und Icons	24
3.3 Usability.....	25
3.3.1 Definition	25
3.3.2 Regeln des UI-Designs.....	25
3.3.3 Farben und Sehen	26
3.3.4 Gestaltungsgesetze der Wahrnehmung.....	27

4	Ist-Zustand	29
4.1	Kooperationspartner.....	29
4.1.1	Bundesagentur für Arbeit.....	29
4.1.2	Automobilhersteller.....	30
4.2	ITOA Tools.....	31
4.2.1	ABCH	32
4.2.2	Acrolinx.....	33
4.2.3	ExtraHop	34
4.2.4	Grafana.....	35
4.2.5	Kibana.....	36
4.2.6	Munin.....	37
4.2.7	Nagios	38
4.2.8	Oracle IT Analytics	39
4.2.9	Splunk.....	40
4.2.10	TrueSight.....	41
4.2.11	Windows Admin Center	42
4.2.12	Windows Capacity Check.....	43
5	Anforderungen an die Neukonzeption	44
5.1	Anforderungen aus ITOA.....	44
5.2	Anforderungen der Benutzer.....	45
5.3	Parameter	47
5.4	Anforderungsliste	48
6	Zwischenfazit	51
6.1	Notwendigkeit einer Neukonzeption.....	51
6.2	Vorteile einer dreidimensionalen Visualisierung	52
7	Konzeption des neuen Systems.....	53
7.1	Allgemeiner Aufbau	53
7.2	Ansichten	54
7.2.1	Dashboard.....	54
7.2.2	Parameter verwalten.....	55
7.2.3	Bericht erstellen	57
7.2.4	Datenquelle ändern.....	57
7.2.5	Einstellungen.....	58
7.3	Panels	60
7.3.1	3D-Ansicht.....	60
7.3.2	2D-Ansicht	64
7.3.3	Systemstatus.....	65
7.3.4	Kennzahlen.....	67
7.3.5	Parameter-Übersicht	68

7.4	Weitere Elemente	70
7.4.1	Filter.....	70
7.4.2	Detailstufenzoom.....	71
7.4.3	Datumsauswahl	71
7.4.4	Prognose.....	74
7.4.5	Detaileinstellungen	74
7.5	Funktionalität.....	75
7.5.1	Bedienbarkeit mit Tastatur und Maus.....	75
7.5.2	Panels verschieben.....	75
7.5.3	Zoomen und Schwenken.....	75
7.5.4	Fokus.....	76
7.6	Styleguide	76
7.6.1	Allgemeiner Aufbau	77
7.6.2	Buttons und Icons	78
7.6.3	Schriftgrößen und -arten	80
7.6.4	Farben	80
8	Prototyp des neuen Systems	82
9	Evaluation.....	83
9.1	Erfüllung der Anforderungen	83
9.2	Verbesserungspotenziale	87
9.3	Fazit.....	88
10	Zusammenfassung und Diskussion	89
10.1	Zusammenfassung.....	89
10.2	Diskussion	91
11	Ausblick.....	93
11.1	Technische Umsetzung.....	93
11.1.1	Allgemeine Hinweise	93
11.1.2	Hinweise für die Tools der Kooperationspartner	94
11.2	Grenzen.....	94
11.2.1	Fachliches Konzept	95
11.2.2	Anzahl gleichzeitiger Parameter.....	95
11.3	Mögliche Erweiterungen.....	95
11.3.1	Mobiler Einsatz des Systems	95
11.3.2	Anzahl gleichzeitiger Parameter.....	96
11.3.3	Dark Mode	96
	Literaturverzeichnis	97
	Anhang	104
	Eidesstattliche Erklärung	105

Abkürzungsverzeichnis

Avg	Durchschnitt (average)
BA	Bundesagentur für Arbeit
ITOA	IT-Operations Analytics
Max	Maximum
Min	Minimum
ML	Maschinelles Lernen (Machine Learning)
RZ	Rechenzentrum / Rechenzentren
UI	User Interface (Benutzerschnittstelle /-oberfläche)
GUI	Graphical User Interface (grafische Benutzerschnittstelle /-oberfläche)

Abbildungsverzeichnis

Abbildung 1: Agiles Vorgehensmodell.....	11
Abbildung 2: Zusammenhang von RZ, Clustern und Servern	19
Abbildung 3: Screenshot Kibana – Bundesagentur für Arbeit	29
Abbildung 4: Screenshot ABCH	32
Abbildung 5: Screenshot Acrolinx.....	33
Abbildung 6: Screenshot ExtraHop.....	34
Abbildung 7: Screenshot Grafana	35
Abbildung 8: Screenshot Kibana.....	36
Abbildung 9: Screenshot Munin.....	37
Abbildung 10: Screenshot Nagios.....	38
Abbildung 11: Screenshot Oracle IT Analytics	39
Abbildung 12: Screenshot Splunk	40
Abbildung 13: Screenshot TrueSight	41
Abbildung 14: Screenshot Windows Admin Center	42
Abbildung 15: Screenshot Windows Capacity Check	43
Abbildung 16: Bereiche des Systems	53
Abbildung 17: Mockup – Dashboard	54
Abbildung 18: Mockup – Parameter verwalten	56
Abbildung 19: Mockup – Parameter verwalten – Detaileinstellungen	56
Abbildung 20: Mockup – Bericht erstellen.....	57
Abbildung 21: Mockup – Datenquelle ändern	58
Abbildung 22: Mockup – Einstellungen	58
Abbildung 23: Mockup – 3D-Ansicht.....	60
Abbildung 24: Mockup – 3D-Ansicht – Detaileinstellungen.....	62
Abbildung 25: Mockup – 2D-Ansicht.....	64
Abbildung 26: Mockup – 2D-Ansicht – Detaileinstellungen.....	64
Abbildung 27: Mockup – Systemstatus	65
Abbildung 28: Mockup – Systemstatus – Detaileinstellungen	66
Abbildung 29: Mockup – Kennzahlen	67
Abbildung 30: Mockup – Kennzahlen – Detaileinstellungen	68

Abbildung 31: Mockup – Parameter-Übersicht	69
Abbildung 32: Mockup – Parameter-Übersicht – Detaileinstellungen	69
Abbildung 33: Mockup – Schnelleinstellungen	70
Abbildung 34: Mockup – Filter	70
Abbildung 35: Mockup – Detailstufenzoom	71
Abbildung 36: Mockup – Datumsauswahl – Schnellauswahl.....	72
Abbildung 37: Mockup – Datumsauswahl – komplexe Auswahl.....	73
Abbildung 38: Mockup – 2D-Ansicht – Prognose.....	74
Abbildung 39: Dashboard – Größenübersicht grob	77
Abbildung 40: Dashboard – Größenübersicht detailliert.....	78
Abbildung 41: Prototyp	82

Tabellenverzeichnis

Tabelle 1: Suchbegriffe und Quellen.....	15
Tabelle 2: Dialogarten.....	23
Tabelle 3: Regeln des UI-Designs.....	26
Tabelle 4: Gestaltungsgesetze der Wahrnehmung.....	28
Tabelle 5: Anforderungen aus ITOA	44
Tabelle 6: Anforderungen der Benutzer – Kernanforderungen	45
Tabelle 7: Anforderungen der Benutzer – Funktionalität.....	46
Tabelle 8: Parameter	47
Tabelle 9: Anforderungsliste	50
Tabelle 10: Toolübersicht (gruppiert).....	51
Tabelle 11: Übersicht über Bereiche und Panels des Systems.....	54
Tabelle 12: Übersicht über Detaileinstellungen-Seitenleisten.....	74
Tabelle 13: Zoomen und Schwenken	76
Tabelle 14: Icons	79
Tabelle 15: Schriftgrößen	80
Tabelle 16: Farben – Parameter.....	80
Tabelle 17: Farben – Balken der 3D-Ansicht	81
Tabelle 18: Evaluation	86
Tabelle 19: Verbesserungspotenziale.....	88
Tabelle 20: Anforderungen aus ITOA – Sicherheit und Architektur.....	104

1 Einleitung

In diesem Kapitel werden zunächst Problem- und Zielstellung sowie die Forschungsfragen dargelegt. Darauf folgt die Beschreibung des methodischen Vorgehens und der Gliederung.

1.1 Problemstellung

Die Menge und Wichtigkeit von Daten nimmt seit dem Fortschreiten der Digitalisierung stets zu. Es werden nicht nur mehr Daten erhoben, sondern sie werden auch immer stärkerer Teil des Wertschöpfungsprozesses [Digi2018]. Um die Menge an Daten verarbeiten und auswerten zu können, werden immer größere und leistungsfähigere Rechenzentren benötigt. Da Unternehmen im Falle eines Ausfalls dieser Datenverarbeitung meistens hohe Schäden erleiden, ist es essenziell, die Verfügbarkeit der Rechenzentren zu gewährleisten [Rüdi2019]. Eine der wichtigsten Maßnahmen ist es hierbei, die Daten der Rechenzentren zu analysieren. Eine solche Analyse wird zwar technisch unterstützt, letztendlich aber von Menschen durchgeführt. Da diese abstrakte Daten nicht so gut verarbeiten können wie Maschinen, müssen die Daten visualisiert werden. Klassischerweise erfolgt diese Visualisierung durch Tabellen sowie durch zweidimensionale Diagramme und erfordert meist einen Wechsel zwischen verschiedenen Tools und innerhalb dieser zwischen mehreren Ansichten. Nur wenige Tools bieten genügend Flexibilität, die Oberfläche nach eigenen Wünschen zu verändern. Für diese Tools wird jedoch ein Konzept benötigt, wie eine Oberfläche zur Überwachung von Rechenzentren gestaltet werden sollte [CSB+2019, Maya2016].

1.2 Zielstellung

Um die beschriebenen Probleme aufzuheben, soll im Rahmen dieser Arbeit prototypisch ein IT-Operations Analytics (ITOA) System konzipiert werden, das die Auslastungsdaten von Rechenzentren auf einen Blick darstellen kann. Dieses System soll die Benutzer bei der Erkennung von abnormalem Verhalten sowie bei der Klassifizierung und Lokalisierung von Problemen unterstützen.

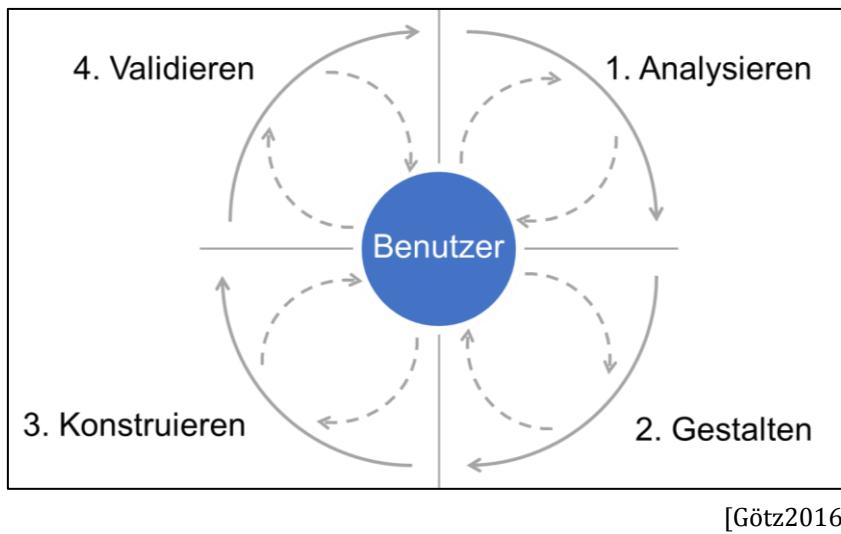
Dazu soll folgenden Forschungsfragen nachgegangen werden:

1. Welche Tools existieren im Umfeld von IT-Operations Analytics?
2. Inwiefern sind existierende Tools im Umfeld von IT-Operations Analytics gegenüber heutigen Anforderungen unzureichend?
3. Wie lässt sich die Lücke zwischen existierenden Tools im Umfeld von IT-Operations Analytics und heutigen Anforderungen aufheben?

Diese Arbeit soll die Lücke zwischen existierenden ITOA-Tools und heutigen Anforderungen schließen. Die Lücke besteht darin, dass existierende Tools mit vorgegebener Oberfläche es nicht ermöglichen, die gesamte Serverlandschaft eines Unternehmens auf einen Blick darzustellen, bzw. dass für existierende Tools, die dazu in der Lage wären, kein entsprechendes Konzept vorhanden ist (vgl. Forschungsfrage 2). Im Bereich der Praxis können durch Schließen der genannten Lücke das Reporting sowie Monitoring und Fehlerüberwachung verbessert werden. Außerdem erleichtert ein entsprechendes System die Erkennung komplexer Fehlerverhalten [Maya2016].

1.3 Methodisches Vorgehen

Im Rahmen dieser Arbeit wurde ein agiles Vorgehensmodell mit vier Phasen angewendet. Mittels dieser können die Schritte zur Konzeption des prototypischen Systems sowie die verwendeten Methoden und Werkzeuge nachvollzogen werden (Abbildung 1).



[Götz2016]

Abbildung 1: Agiles Vorgehensmodell

Im Folgenden werden die einzelnen Phasen im Detail erläutert. Die theoretische Basis zu den verwendeten Methoden und Werkzeugen sowie zu dem Vorgehensmodell wird in Kapitel 2 erläutert.

Phase 1: Analysieren (Ist-Zustand und Anforderungen)

In der ersten Phase wurden mittels Literaturrecherche existierende Ansätze und Systeme im Bereich ITOA analysiert. Anschließend wurden Anforderungen an das System gesammelt und zu einer Anforderungsliste aggregiert. Dazu wurde eine Kombination aus Literaturrecherche, Experteninterviews und Brainstorming angewandt. Diese Methoden wurden kombiniert, um Anforderungen aus der Wissenschaft und der Praxis zu berücksichtigen und so eine umfassende Anforderungsliste zu erhalten. Konkret wurden zunächst die ersten beiden Methoden parallel angewandt. Daraus wurde eine vorläufige Liste erstellt, die dann mit den Benutzern mittels Brainstorming spezifiziert und erweitert wurde.

Phase 2: Gestalten (Mockups)

In der zweiten Phase wurden auf Basis der Anforderungsliste und der bestehenden Tools erste Mockups erstellt. Diese wurden in Abstimmung mit den Benutzern angepasst und erweitert. Die als Ergebnis der zweiten Phase entstandenen Mockups stellen die wesentlichen Elemente und Funktionen des Systems dar.

Phase 3: Konstruieren (Prototyp)

Auf Basis der Mockups wurde in der dritten Phase Prototyping angewandt. Dazu wurden die Schaltflächen der Mockups zu einem klickbaren Prototypen verlinkt. Dieser wurde den Benutzern zum Testen vorgelegt, um somit Feedback zu erhalten und anschließend berücksichtigen zu können.

Phase 4: Validieren (Evaluation)

In der letzten Phase wurde der Prototyp mit der Anforderungsliste abgeglichen und den Benutzern zum Testen zur Verfügung gestellt. Dazu wurde zunächst beschrieben, durch welche Elemente oder Funktionen des Systems die einzelnen Anforderungen erfüllt werden. Anschließend konnten die Benutzer den Erfüllungsgrad der einzelnen Anforderungen bewerten. Diese Bewertungen wurden aggregiert und es wurde eine Tabelle mit Verbesserungspotenzialen erstellt.

In der Evaluationsphase ist zu beachten, dass die Validation primär auf theoretischer Ebene erfolgte, da für die Tests lediglich das prototypische System vorhanden war. Um möglichst zuverlässige Ergebnisse zu erzielen, wurden die Benutzer aufgefordert den Prototyp zu ‚testen‘. Würde man die Benutzer auffordern, den Prototyp zu ‚validieren‘, wären diese beeinflusst und würden über kleinere Fehler hinwegsehen [Pern2017].

Weitere Durchläufe

In den Beschreibungen der vorherigen Phasen wurde jeweils dasjenige Vorgehen gebündelt beschrieben, das in mehreren Durchläufen dort angewandt wurde. In der Praxis wurden alle Phasen mehrfach durchlaufen. Dabei wurden weitere Anforderungen aufgenommen, zusätzliche Mockups erstellt und der Prototyp überarbeitet. Als Ergebnis aller Durchläufe ist der in Kapitel 8 dargestellte und erfolgreich evaluierte Prototyp entstanden.

1.4 Gliederung der Arbeit

Innerhalb dieser Arbeit werden zunächst verwendete Methoden und Werkzeuge vorgestellt sowie grundlegende Begriffe erklärt. Im Anschluss werden in je einem Kapitel die Phasen des Vorgehensmodells durchlaufen: Zuerst erfolgt die Analyse des Ist-Zustandes im Bereich ITOA, woraus die Beantwortung der ersten Forschungsfrage resultiert. Anschließend werden Anforderungen an ITOA-Systeme erhoben und zur Beantwortung der zweiten Forschungsfrage mit den Systemen des Ist-Zustandes abgeglichen. Die dabei identifizierten Lücken dienen als Basis für die Konzeption eines neuen Systems. Dieses Konzept beantwortet gleichzeitig die letzte Forschungsfrage. Abschließend folgt eine Evaluation des prototypisch konzipierten Systems zusammen mit den Benutzern und auf Basis der erhobenen Anforderungen. Im Ausblick werden Hinweise für die technische Umsetzung gegeben sowie Grenzen und mögliche Erweiterungen aufgezeigt.

1.5 Anmerkungen zur Schreibweise

Benutzer

Wenn im Rahmen dieser Arbeit der Begriff Benutzer verwendet wird, sind damit die zukünftigen Benutzerinnen und Benutzer des Systems sowie die Benutzerinnen und Benutzer derzeit eingesetzter Systeme gemeint. Primär sind dies die entsprechenden Mitarbeitenden der Kooperationspartner.

Kooperationspartner

Unter dem Begriff Kooperationspartner werden im Folgenden die Bundesagentur für Arbeit und der Automobilhersteller zusammengefasst, deren Benutzer im Zentrum des Vorgehensmodells stehen. Diese standen für Interviews zur Verfügung, haben Anforderungen beigetragen sowie den Prototypen getestet und evaluiert.

2 Verwendete Methoden und Werkzeuge

In diesem Kapitel werden die im Rahmen dieser Arbeit verwendeten Methoden und Werkzeuge beschrieben. Dazu erfolgt jeweils zunächst eine allgemeine Beschreibung und anschließend ein Absatz zum Einsatz in dieser Arbeit.

2.1 Vorgehensmodell

Beim klassischen Wasserfallmodell werden nacheinander die Phasen Analyse, Entwurf, Implementierung, Test und Wartung durchlaufen [Royc1970]. Innerhalb dieser Arbeit wird in der Analysephase der Ist-Zustand erfasst. In den Phasen Entwurf und Implementierung werden Mockups und ein Prototyp erstellt. In der Testphase sehen die Benutzer den Prototypen und geben erstmalig Feedback. Dieses muss dann in einem Rückschritt in der Entwurfsphase berücksichtigt werden. Da die technische Umsetzung nicht Teil dieser Arbeit ist, entfällt die Wartungsphase.

In der Praxis sind die einzelnen Phasen oft nicht klar trennbar und mindestens nach der Testphase sind Rückschritte zu vorherigen Phasen erforderlich. Die späte Einbindung der Benutzer hat zur Folge, dass falsche Annahmen bei der Analyse erst spät erkannt werden, wodurch der gesamte Prozess die Bedürfnisse der Benutzer nicht gänzlich berücksichtigt [Royc1970].

Das zuvor dargestellte agile Vorgehensmodell (Abbildung 1) besitzt ähnliche Phasen wie das Wasserfallmodell, integriert jedoch in jeder davon die Benutzer. Im ersten Durchlauf der Phasen können grobe Mockups erstellt und in weiteren Durchläufen können diese näher spezifiziert werden. Durch den Einsatz mehrerer Durchläufe wird das Design Schritt für Schritt präziser und der Prototyp detaillierter [Götz2016]. Insgesamt eignet sich das agile Vorgehensmodell insbesondere durch die starke Integration der Benutzer für die Konzeption des ITOA-Systems. Aufgrund dessen wird das agile Vorgehensmodell dem Wasserfallmodell vorgezogen.

2.2 Literaturrecherche

Im Rahmen der Literaturrecherche wurde eine Kombination aus der Methode der konzentrischen Kreise und der systematischen Suche angewandt. Diese beiden Methoden werden im Folgenden zunächst einzeln beschrieben. Anschließend wird auf den konkreten Einsatz in dieser Arbeit eingegangen.

Bei der Methode der konzentrischen Kreise wird ein zentrales Werk herangezogen, aus dessen Literaturverzeichnis relevante Quellen gesucht werden. Im nächsten Schritt wird in deren Quellen nach weiterer Literatur gesucht. Die Anzahl an relevanten Quellen wächst dabei solange an, bis sich die Quellen wiederholen. Bei dieser Methode gilt es jedoch zu beachten, dass sich Gruppen von Autoren teilweise nur gegenseitig zitieren, wodurch man schnell keine neuen Quellen findet. Darüber hinaus kann es sein, dass sich Texte auf ältere Veröffentlichungen beziehen, die sich wiederum auf noch ältere Werke beziehen, wodurch man nur stark veraltete Quellen erhält. Daher bietet es sich an, die Methode der konzentrischen Kreise mit anderen Methoden zu kombinieren [Korn2016].

Bei der Methode der systematischen Suche werden aus der Themenstellung Schlüsselwörter abgeleitet, nach denen gezielt gesucht wird. Dieses Vorgehen eignet sich insbesondere für die Suche nach Einzelinformationen [Korn2016].

Zur Erfassung des Ist-Zustandes und zur Ermittlung der Anforderungen wurde mittels einer systematischen Suche im Web nach den unten genannten Suchbegriffen in den darunter stehenden Quellen gesucht (Tabelle 1). Dazu wurden die Suchbegriffe der angegebenen Mengen mehrfach kombiniert (z. B. ,Data Center Visualization', ,Visualisierung Server Auslastung' oder ,ITOA Tools'). Die dabei gefundenen Literaturverzeichnisse wurden nach der Methode der konzentrischen Kreise für weitere Recherche verwendet.

Suchbegriffe	{Server, Rechenzentrum, Data Center, CPU}, {Load, Lastverteilung Auslastung, Utilization}, {Visualisierung, Visualization}, {Netzwerk Monitoring}, {ITOA, IT-Operations Analytics}, {Tool, System, Software}
Quellen	Google Scholar, Google Websuche, Google Bildersuche, Literaturverzeichnisse bereits gefundener Paper/Tools

Tabelle 1: Suchbegriffe und Quellen

2.3 Experteninterviews

In der Literatur werden zahlreiche Interviewformen und Ausprägungen beschrieben. Häufig gliedert man diese nach ihrem Standardisierungsgrad. Dabei werden von Flick et al. und Lamnek folgende Typen unterschieden [FKK+1995, Lamn2016]:

1. Stark standardisiertes Interview (Fragebogenerhebung)
2. Gering standardisiertes Interview (Leitfaden-Interview)
3. Offenes Interview (freie Erzählung nach spezifischer Erzählaufforderung)
4. Ethnografisches Interview

Im Folgenden wird das Leitfaden-Interview näher betrachtet und es wird begründet, inwiefern es sich für die vorliegende Arbeit eignet.

Leitfaden-Interviews ermöglichen durch die geringe Standardisierung offene Frageformulierungen und bieten Flexibilität in der Reihenfolge und Detaillierung der Fragen. Diese Flexibilität erleichtert es, den Ist-Zustand bezüglich der Systeme der Kooperationspartner sowie die Anforderungen an solche Systeme zu erheben. Im Interview kann direkt auf individuelle Bedürfnisse und Ideen eingegangen werden [FKK+1995, Lamn2016].

Eine Ausprägung des Leitfaden-Interviews ist das leitfadengestützte Experteninterview. Dieses unterscheidet sich vom klassischen Leitfaden-Interview darin, dass eine spezielle Personengruppe befragt wird. Befragt werden dabei Experten [GILa2010]. In dieser Arbeit sind das die Benutzer der Kooperationspartner, welche teilweise seit vielen Jahren mit den derzeit eingesetzten Tools arbeiten und somit umfassende Kenntnisse in diesem Bereich besitzen. Aufgrund dieser können die Benutzer qualitativ hochwertige Anforderungen aus der Praxis beitragen.

Aufgrund der Flexibilität in Kombination mit einem Mindestmaß an Struktur werden in dieser Arbeit Leitfaden-Interviews durchgeführt. Um möglichst praxisnahe Informationen zu erhalten, werden dazu ausschließlich Experten befragt, wodurch konkret leitfadengestützte Experteninterviews durchgeführt werden.

2.4 Brainstorming

Als eine Kreativitätstechnik dient Brainstorming dazu, in einer Gruppe von mehreren Personen Ideen zu sammeln und Lösungen zu erarbeiten. Dabei sollten nach Clark folgende Rahmenbedingungen eingehalten werden [Clar1972]:

- Gruppengröße: Bestenfalls 4 – 20 Personen.
- Leitung: Sollte neutral sein und die Aussagen der Gruppe nicht lenken.
- Protokollführer: Sollte neutral sein oder kann Teil der Gruppe sein.
- Quantität: Fokus auf Quanität, Qualität ist zweitrangig.

In dieser Arbeit wurde Brainstorming dazu eingesetzt, um zusätzlich zu den Interviews weitere Anforderungen zu identifizieren und Verbesserungsvorschläge für die Mockups zu erhalten. Dies erfolgte mehrfach jeweils mit vier Personen und mit beiden Kooperationspartnern.

2.5 Prototyping

Als Methode der Softwareentwicklung dient Prototyping bzw. Prototypenbau dazu, schnell erste Ergebnisse zu erzielen und daraus frühzeitiges Feedback zu generieren. Dabei werden früh Modelle erstellt, die mit den Kunden getestet werden können [Empr2019, Wiki2019]. Im Folgenden werden einige Methoden für das Prototyping vorgestellt und jeweils kurz beschrieben. Eine detaillierte Beschreibung dieser Methoden findet sich im Blogartikel von Emprechtiner [Empr2019]:

- Skizzen und Diagramme: Erstellung einfacher Illustrationen.
- Storyboards: Darstellung der Customer Journey in einer Reihe von Bildern.
- Role Play: Nachspielen von Szenen und Situationen.
- Paper Prototyping: Papierprototypen für bildschirmbasierte Produkte.
- Mockups: Modelle ohne Funktionalität, die wie fertiges Produkt wirken.
- Wireframes: Gröbere Form der Mockups, Skelett des Userinterfaces.
- Cardboard Prototyping: Erstellung dreidimensionaler Modelle aus Pappe.
- Prototyping mit Erklär-Videos: Erklärung der geplanten Funktionalität.
- Wizard of Oz Prototyping: Nachahmen/Faken von Funktionen.

Für die vorliegende Arbeit eignen sich am besten Mockups und Wireframes. Da es sich bei Mockups um eine komplexere Darstellung handelt, die den Benutzern mehr Details zeigen kann, werden diese im Konzeptionskapitel eingesetzt. Sie zeigen ebenso wie Wireframes den Aufbau von Menüs und Ansichten. Darüber hinaus ermöglichen es die Mockups jedoch, einzelne Funktionen und Bedienelemente darzustellen. Über die statischen Mockups hinaus wird in dieser Arbeit ein klickbarer Prototyp erstellt. Dieser besteht aus verlinkten Mockups, die jeweils einen spezifischen Zustand des Systems darstellen. Dadurch erhalten die Benutzer einen Eindruck über die Funktionalität des Systems, ohne dass dazu ein System programmiert werden muss.

2.6 Visualisierungstool für die Mockups

Für die Erstellung der Mockups und des Prototypen wird Microsoft PowerPoint verwendet. PowerPoint bietet einen großen Funktionsumfang und ist insbesondere nützlich beim Designen von einzelnen Elementen, bei gleichmäßiger Ausrichtung von Elementen und bei der Erstellung von Verlinkungen. Zudem ist PowerPoint stark verbreitet und somit bei vielen Benutzern bekannt. Die Mockups könnten daher leicht von Dritten bearbeitet oder ergänzt werden.

Beim Vergleich anderer Tools wurde sich aus finanziellen Gründen gegen kommerzielle Tools entschieden. Bekannte Tools wie moqups.com oder Balsamiq benötigen einen höheren Einarbeitungsaufwand und bieten keinen signifikanten Vorteil gegenüber PowerPoint, sodass sich auch gegen diese entschieden wurde.

Aus den Mockups wird ein klickbarer Prototyp erstellt, indem einzelne Slides verlinkt werden. Somit erhält der Benutzer einen Eindruck über die Funktionalität des späteren Systems.

3 Grundlagen

In den folgenden Unterkapiteln werden zunächst allgemeine Begriffe und anschließend Begriffe bezüglich GUI, Steuerelementen, Usability sowie Farben, Formen und Symbolen erläutert.

3.1 Begriffsdefinitionen – allgemein

3.1.1 Verbundstrukturen im Rechenzentrum

Die untenstehende Abbildung zeigt den Zusammenhang von Rechenzentren, Clustern und Servern am Beispiel der Bundesagentur für Arbeit (Abbildung 2).

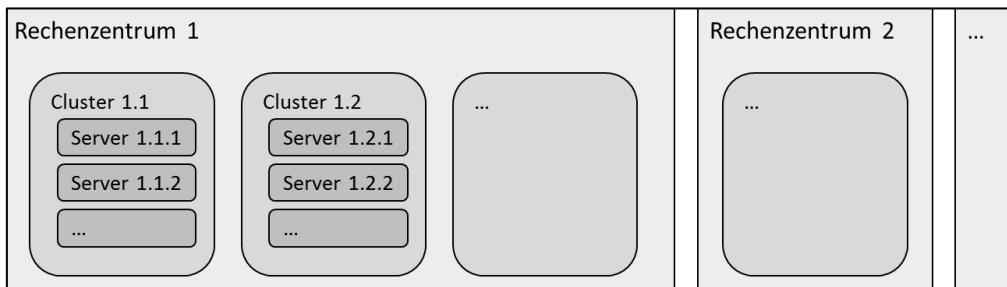


Abbildung 2: Zusammenhang von RZ, Clustern und Servern

Ein *Rechenzentrum* (RZ) bezeichnet ein Gebäude bzw. die Räumlichkeiten, in welchen sich zentralisiert die Rechentechnik einer oder mehrerer Unternehmen oder Organisationen befindet. Dazu zählen sowohl Server als auch die nötige Infrastruktur wie Stromversorgung und Kühlung [ArPo2004].

Jedes RZ beinhaltet eine Vielzahl an *Servern*, also einzelne Computer (auch *Knoten* genannt). Diese Server können Dienstprogramme, Daten oder andere Ressourcen bereitstellen. In der Theorie unterscheidet man zwischen Software- und Hardware-Servern. Erstere beziehen sich auf die ursprüngliche Bedeutung, dass Server Dienste anbieten. Unter einem Hardware-Server versteht man den Computer, auf dem diese Dienste angeboten werden. In der Praxis sind meistens beide Begriffe zugleich gemeint, wenn von ‚Server‘ gesprochen wird. In diesem Sinne wird der Begriff auch im Rahmen dieser Arbeit verwendet [OsMi2016].

Die einzelnen Server in einem RZ sind zu *Clustern* zusammengeschlossen. Ein Cluster im Sinne der Informationstechnologie ist somit ein Verbund aus mehreren Computern [DrOs2017].

3.1.2 Serverauslastung

„Die Serverauslastung ist ein Prozentwert, der darüber Aufschluss gibt, in welchem Maße die Server-Infrastruktur in Anspruch genommen wird“ [Ryte2014] und ist abhängig von der Übertragungsbandbreite, der Anzahl an Anfragen, der Art und Größe des zu ladenden Contents und der Serverarchitektur [Ryte2014]. Je höher die Auslastung ist, desto geringer sind die Kosten der Cloud Infrastruktur. Dabei muss die Einsparung gegen das Risiko abgewogen werden, dass kurzfristig mehr Anfragen gestellt werden, als beantwortet werden können [WuBu2015].

In der vorliegenden Arbeit ist die Serverauslastung der wichtigste Parameter bei der Analyse der Server und Rechenzentren.

3.1.3 Wireframe, Mockup und Prototyp

Ein Wireframe stellt den Inhalt einer Webseite, die Struktur von Informationen und eine Beschreibung der möglichen Interaktionen dar. Dazu werden nur wenige Farben und keine konkreten Inhalte, Bilder oder Icons verwendet. Da Wireframes auf Grund der geringeren Komplexität schneller erstellt werden können, eignen sie sich als Grundgerüst für das finale Design [Beck2016]. Folgend wird ein Wireframe für die Darstellung des allgemeinen Aufbaus verwendet.

Ein Mockup ist ein digitaler Entwurf einer Webseite und stellt das Design dar. Mockups enthalten Farben, Icons und Inhalte und dienen dem Grafiker dazu, schnell Feedback zur Optik einzuholen und auf Basis dessen Änderungen vorzunehmen [Beck2016]. Dazu werden zahlreiche Mockups verwendet, um den konkreten Aufbau aller Ansichten, Panels und Interaktionsmöglichkeiten darzustellen.

Ein Prototyp ist eine nahezu fertige Abbildung des Produktes und stellt die Benutzeroberfläche sowie die möglichen Interaktionen dar. Dabei liegt der Fokus auf der Interaktion und der Simulation von Funktionen, ein Prototyp muss nicht programmiert werden. [Beck2016]. In der vorliegenden Arbeit werden die Schaltflächen der Mockups so verlinkt, dass daraus ein klickbarer Prototyp entsteht. Dadurch sieht der Benutzer, wie das finale Produkt funktionieren soll und gleichzeitig bleibt die Designkomplexität der Mockups erhalten.

3.1.4 IT-Operations Analytics

IT-Operations Analytics (ITOA) wird nach Gartner definiert als „Technologie oder Service, der aus großen Mengen an IT-Betriebsdaten deduktive und/oder induktive Inferenzen (Schlussfolgerungen) zieht und diese dann in Form von Dashboards oder Reports darstellt und speichert.“ [DeSc2013]. Ziel ist es, der IT zu ermöglichen, einen Wertbeitrag für das Unternehmen zu generieren [DeSc2013].

Auf dem Markt werden viele ITOA-Tools angeboten, die den Benutzer von der Verwaltung und Erhebung der Daten bis zur Visualisierung der aus den Daten generierten Schlussfolgerungen unterstützen (vgl. Kapitel 4). Das im Rahmen dieser Arbeit zu konzipierende ITOA-System soll den Benutzern ermöglichen, Fehler und Anomalien zu bemerken, zu lokalisieren und zu klassifizieren.

3.1.5 Maschinelles Lernen

Maschinelles Lernen (Machine Learning, ML) ist eine Form der künstlichen Intelligenz, bei der Maschinen und Programme auf Grundlage von beispielhaften Vergangenheits- und Vergleichsdaten eigenständig Muster erkennen und darauf aufbauend Entscheidungen treffen bzw. agieren können [Pous2018].

Im zu konzipierenden System soll ML zur Fehlererkennung und -prognose eingesetzt werden. Dazu eignet sich die Methode der Zeitreihenanalyse. Bei dieser analysiert ein ML-Algorithmus die verfügbaren Daten auf Abhängigkeiten und Veränderungen über die Zeit. Die Daten müssen dazu in einem empirischen Zeitverlauf vorliegen. Durch die Zeitreihenanalyse können Prognosen der Zeitreihenverläufe in der Zukunft berechnet und Trends erkannt werden [Kamp2018].

In der vorliegenden Konzeption sollen Zeitreihenanalysen auf den Auslastungsdaten der Rechenzentren durchgeführt werden. Bei der Analyse sollen alle verfügbaren Parameter im Rahmen der Anwendungsfälle ‚Fehlererkennung‘ und ‚Prognose‘ berücksichtigt werden.

Fehlererkennung

Falls bereits ein Fehler aufgetreten ist, soll im Systemstatus-Panel angezeigt werden, wo sich der Fehler voraussichtlich befindet, wann genau er aufgetreten ist und mit welcher Wahrscheinlichkeit der Algorithmus den Fehler entsprechend klassifiziert hat.

Prognose

Um Fehler oder Anomalien bereits im Vorfeld erkennen und dadurch ggf. vermeiden zu können, soll der ML-Algorithmus vorhersagen, wo welche Art von Anomalie mit welcher Wahrscheinlichkeit auftreten wird.

Im Rahmen dieser Arbeit werden die ML-spezifischen Elemente der Oberfläche konzipiert, die Details bezüglich des Algorithmus sind nicht Teil der Arbeit.

3.2 GUI und Steuerelemente

Wenn der Begriff ‚User Interface‘ (UI) benutzt wird, ist meistens die grafische Benutzerschnittstelle / -oberfläche (Graphical User Interface, GUI) gemeint. Eine andere Form der Benutzeroberfläche ist zum Beispiel das textbasierte Interface. Eine GUI bildet die Schnittstelle zwischen Mensch und Maschine. Dabei wird das Ziel verfolgt, dem Menschen in möglichst optimierter Form die Kontrolle über die Maschine zu geben und Ausgaben der Maschine für den Menschen lesbar darzustellen [RaAu2017].

Zur Interaktion mit einer GUI verwendet der Benutzer Steuerelemente (auch Bedien- oder Interaktionselemente genannt). Zu diesen zählen beispielsweise Buttons, Textfelder und Slider. Eine umfassende Liste an Steuerelementen kann in der DIN EN ISO 9241-143 nachgelesen werden [ISO2012].

In den folgenden Kapiteln werden speziellere GUI- und Steuerelemente erläutert.

3.2.1 Grundsätze der Dialoggestaltung

In Fenstern, sogenannten Dialogen, können Steuerelemente angezeigt werden, die auf den ersten Blick nicht sichtbar sind. Die Dialoge werden klassischerweise in Primär- und Sekundärdialoge geteilt. Beide Formen können jeweils modal oder nicht-modal sein. In untenstehender Tabelle werden die Dialogarten kurz definiert und es wird an einem Beispiel gezeigt, wo die jeweilige Dialogart in der Konzeption eingesetzt wird (Tabelle 12).

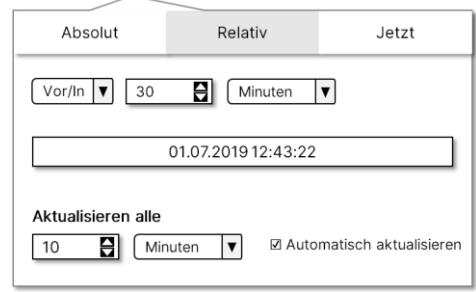
Begriff	Definition	Beispiel aus dieser Arbeit
Primärdialog	Dient der direkten Aufgabenerfüllung und wird geschlossen, wenn die Aufgabe erledigt ist [Riew2010].	Auswahl der Art der Aggregation 
Sekundärdialog	Bietet situationsabhängig Zusatzinformationen. Wenn der Sekundärdialog geschlossen wird, wird das Primärfenster wieder aktiv [Riew2010].	
Modaler Dialog	Muss beendet sein, bevor eine andere Aufgabe durchgeführt bzw. bevor ein anderes Fenster aktiviert werden kann [Riew2010].	Siehe Primär- und Sekundärdialog.
Nicht-modaler Dialog	Ermöglicht dem Benutzer eine Aufgabe durchzuführen, während das ursprüngliche Fenster geöffnet bleibt [Riew2010].	In dieser Arbeit nicht vorhanden.

Tabelle 2: Dialogarten

3.2.2 Dashboard

Unter einem Dashboard versteht man die auf einer Seite bzw. Bildschirmansicht übersichtlich aufbereitete Darstellung komplexer Inhalte. Durch entsprechende Interaktionselemente kann der Benutzer Änderungen simulieren und so die Auswirkungen auf andere Diagramme oder Kennzahlen beobachten [Cham2018].

3.2.3 Seitenleiste

Eine Seitenleiste (auch Sidebar genannt) bezeichnet den vertikalen Bereich einer Webseite, in dem sich Zusatzinformationen befinden. Man unterscheidet zwischen linker und rechter Seitenleiste. Die linke Seitenleiste wird typischerweise als weiteres Navigationselement verwendet. Da die rechte Seitenleiste aufgrund der Leserichtung erst später wahrgenommen wird, enthält sie meist eher Zusatzinformationen [Zitz2018].

In der Konzeption wird die linke Seitenleiste als Navigationsleiste betrachtet, von der aus zwischen den verschiedenen Ansichten gewechselt werden kann. Die rechte Seitenleiste enthält Detaileinstellungen, die über die allgemeinen Einstellungen in der oberen Leiste hinausgehen.

3.2.4 Tooltip

Tooltips, auch QuickInfo genannt, sind kleine Pop-up-Fenster, die erscheinen, wenn der Mauszeiger kurze Zeit über einem Element verweilt (Mouseover). Diese Pop-up-Fenster enthalten meist kurze Beschreibungen bezüglich des anvisierten Elements. Klassischerweise wird bei einem Link angezeigt, wohin er führt, während bei einem Button angezeigt wird, wie er heißt und was er bewirkt [Self2017].

In dem zu konzipierenden Systems sollen alle Buttons Tooltips enthalten sein, in denen der Name des Buttons und die jeweilige Aktion beschrieben sind.

3.2.5 Buttons und Icons

Ein Button, auch Taste oder Schaltfläche genannt, ist meist mit einem Text versehen und löst eine bestimmte Aktion aus, wenn er angeklickt wird [Garr2010].

Ein Icon ist „eine visuelle Darstellung eines Objektes, einer Aktion oder einer Idee“ [Bedf2014]. Icons haben unter anderem die Vorteile, dass sie einfach angeklickt und schnell erkannt werden können, platzsparend sind, nicht übersetzt werden müssen und fürs Auge ansprechend sein können. Dabei muss jedoch darauf geachtet werden, dass die Benutzer den Zweck der jeweiligen Icons direkt erkennen können [Bedf2014].

3.3 Usability

Eine gute Benutzerfreundlichkeit ist eine der wichtigsten Anforderungen an das zu konzipierende System. Im Folgenden wird der Begriff Usability definiert und es werden Regeln vorgestellt, die ein benutzerfreundliches System erfüllen muss. Im Anschluss werden Aspekte beschrieben, die bei der Farbgebung des Systems berücksichtigt werden müssen und es werden die Gestaltungsgesetze der Wahrnehmung erläutert.

3.3.1 Definition

Nach der DIN EN ISO 9241-11 wird Usability definiert als „das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen.“ [ISO2016].

3.3.2 Regeln des UI-Designs

In der Literatur finden sich verschiedene Sammlungen von Heuristiken, Regeln und Gesetzen für das Design einer Benutzeroberfläche (wie zum Beispiel [Niel1993] und [Chri2015]). Stellvertretend für diese sich ähnelnden Sammlungen werden in der untenstehenden Tabelle die „Golden Rules of User Interface Design“ von Theo Mandel aufgelistet [Mand2012]. Diese decken die wesentlichen Punkte der anderen Sammlungen ab (Tabelle 3).

Die Nummerierung der Regeln entspricht der in der konsolidierten Anforderungsliste (Kapitel 5.4).

3. Usability	
3.1	Gib Benutzern die Kontrolle
3.1.1	Nutze Modi mit Bedacht (gleiche Buttons = gleiche Aktionen)
3.1.2	Ermögliche die Bedienung jeweils mit Tastatur und Maus
3.1.3	Ermögliche Benutzern den Fokus zu ändern
3.1.4	Zeige beschreibende Nachrichten und Texte an
3.1.5	Ermögliche sofortige und reversible Aktionen

3.1.6	Sinnvolle Navigation mit Abbruchmöglichkeiten
3.1.7	Berücksichtige Benutzer verschiedener Qualifikationsstufen
3.1.8	Machen Sie die Benutzeroberfläche transparent (der Benutzer sollte sich jederzeit voll auf seine Aufgabe fokussieren können)
3.1.9	Ermögliche Benutzern das Anpassen der Benutzeroberfläche
3.1.10	Ermögliche Benutzern das direkte Bearbeiten von Objekten der Oberfläche
3.2	Reduziere den Gedächtnisaufwand der Benutzer
3.2.1	Entlaste das Kurzzeitgedächtnis
3.2.2	Unterstütze den Benutzer dabei Elemente wiederzuerkennen (z. B. durch Tooltips oder kontextabhängige Hinweise)
3.2.3	Stelle visuelle Hinweise bereit
3.2.4	Ermögliche Standardeinstellungen sowie Aktionen rückgängig zu machen und zu wiederholen
3.2.5	Stelle Shortcuts bereit
3.2.6	Strebe eine objektorientierte Syntax an
3.2.7	Verwende reale Metaphern
3.2.8	Nutze progressive Offenlegung von Funktionen (zeige dem Benutzer Funktionen nur dann, wenn er sie benötigt)
3.2.9	Strebe visuelle Klarheit an
3.3	Strebe Konsistenz an
3.3.1	Erhalte den Kontext der Aufgaben der Benutzer (Kontinuität)
3.3.2	Stelle Konsistenz zu anderen Systemen her (ähnlicher Aufbau von Menüs und gleiche Funktionsweise von bekannten Buttons)
3.3.3	Gleiche Aktionen sollten zu gleichen Ergebnissen führen
3.3.4	Strebe ein optisch ansprechendes Design an
3.3.5	Ermutige den Benutzer Funktionen auszuprobieren

[Mand2012]

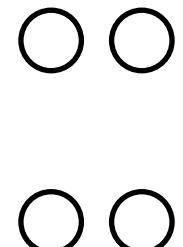
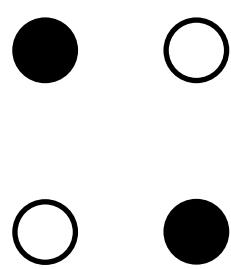
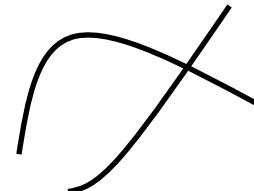
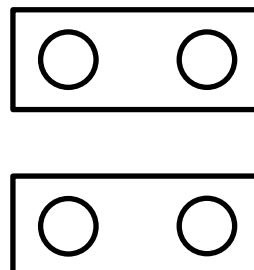
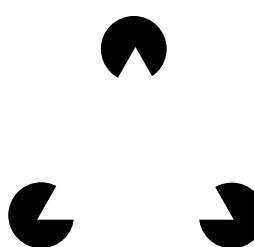
Tabelle 3: Regeln des UI-Designs

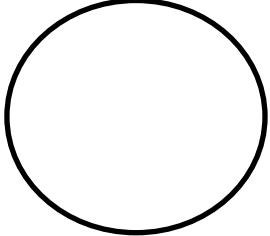
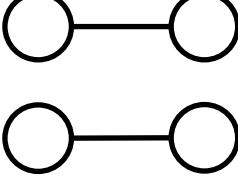
3.3.3 Farben und Sehen

Rund 8% aller Männer und etwa 0,5% aller Frauen sind farbenblind bzw. haben eine Sehschwäche [Like2015]. Daher ist es notwendig, dass der Benutzer ein Interface vollständig ohne Farben bedienen kann. Allgemein sollte darauf geachtet werden, dass keine hohen Kontraste oder stark gesättigten Farben verwendet werden [Niel1993]. Diverse Farbpaletten sind gleichermaßen für Farbenblinde und nicht-Farbenblinde geeignet (vgl. Kapitel 7.6.4).

3.3.4 Gestaltungsgesetze der Wahrnehmung

In untenstehender Tabelle werden die Gestaltungsgesetze nach Wertheimer aufgelistet und anhand von Beispielen jeweils kurz beschrieben (Tabelle 4) [Wert1923].

Gestaltungs-gesetz	Beschreibung	Beispiel
Gesetz der Nähe	Räumlich oder zeitlich benachbarte Elemente werden als zusammengehörig und damit als Figur empfunden. Im Beispiel werden die oberen und unteren Kreise jeweils als zusammengehörig empfunden.	
Gesetz der Gleichartigkeit / Ähnlichkeit	Ähnliche beziehungsweise gleichartige Elemente erscheinen als zusammengehörig, auch wenn sie räumlich getrennt sind. Die Ähnlichkeit kann absteigend gut erreicht werden durch Farbe, Helligkeit, Größe, Orientierung oder Form. Im Beispiel werden die schwarzen und weißen Kreise jeweils als zusammengehörig empfunden.	
Gesetz der guten Fortsetzung	Elemente, die räumlich oder zeitlich in einfacher Folge angeordnet sind, erscheinen als zusammengehörig und damit als Figur. Im Beispiel werden die Linien als Bogen nach unten und als aufsteigende Linie empfunden.	
Gesetz des gemeinsamen Bereichs	Elemente, die von einem Rahmen umgeben sind, werden als zusammengehörig empfunden. Im Beispiel werden die oberen und unteren Kreise jeweils als zusammengehörig empfunden.	
Gesetz der Schließung / Geschlossenheit	Nahezu geschlossene Konturen werden bei der Wahrnehmung gedanklich geschlossen, wobei das Innere zur Figur und das Äußere zum Hintergrund wird. Im Beispiel wird das weiße Dreieck in der Mitte der Symbole weißer empfunden, als der Hintergrund.	

Prinzip der guten Gestalt	<p>Die Gestaltungsprinzipien wirken beim Auswerten einer Grafik immer so zusammen, dass eine möglichst einfache, regelmäßige, symmetrische und geschlossene Figur entsteht.</p> <p>Im Beispiel wird die Ellipse als Kreis wahrgenommen.</p>	
Gesetz der Verbundenheit	<p>Miteinander verbundene Objekte werden als zusammengehörig empfunden.</p> <p>Im Beispiel werden die oberen und unteren Kreise jeweils als zusammengehörig empfunden.</p>	

In Anlehnung an [Wert1923]

Tabelle 4: Gestaltungsgesetze der Wahrnehmung

4 Ist-Zustand

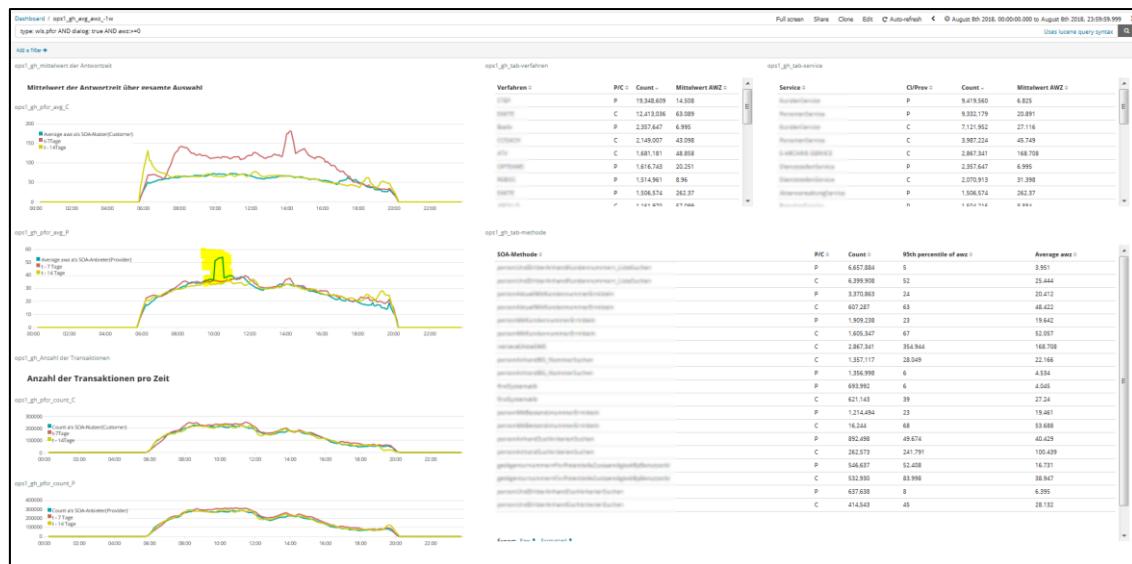
In diesem Kapitel werden zunächst die derzeit eingesetzten Tools der Kooperationspartner vorgestellt. Anschließend werden insgesamt zwölf Tools betrachtet, die auf die Analyse von Serverdaten spezialisiert sind oder die sich für diesen Zweck nutzen lassen. Die Auswahl der Tools basiert auf den Ergebnissen der Literaturrecherche.

4.1 Kooperationspartner

In den folgenden Unterkapiteln soll für die beiden Kooperationspartner jeweils beschrieben werden, welches Monitoring Tool verwendet wird, wie groß die Serverlandschaft ist und wo Optimierungspotenziale liegen.

4.1.1 Bundesagentur für Arbeit

Die Bundesagentur für Arbeit (BA) verwendet für die Überwachung ihrer Rechenzentren das Tool Kibana. Bei der BA werden drei Rechenzentren mit je vier Clustern und insgesamt über 10.000 Servern eingesetzt [CSB+2019].



[CSB+2019]

Abbildung 3: Screenshot Kibana – Bundesagentur für Arbeit

Aus Datenschutzgründen wurde nur ein verpixelter Screenshot bereitgestellt (Abbildung 3). Aus diesem geht hervor, dass bei der Bundesagentur für Arbeit derzeit einfache Darstellungen wie zweidimensionale Liniendiagramme sowie viele Tabellen eingesetzt werden. Dabei müssen sich die Benutzer durch mehrere Ansichten klicken und jeweils verschiedene Detailstufen durchsehen, um Anomalien erkennen zu können. Da es sich bei Kibana um ein vergleichsweise mächtiges Tool handelt (vgl. Kapitel 4.2.5), lässt sich insgesamt festhalten, dass die BA das Potenzial von Kibana nicht ausschöpft.

4.1.2 Automobilhersteller

Aus Datenschutzgründen dürfen an dieser Stelle keine konkreten Aussagen bezüglich Name des Automobilherstellers sowie Aufbau und Größe der Serverlandschaft genannt werden. Im Rahmen der Interviews wurden jedoch nachfolgende Informationen zur Verfügung gestellt:

Der Automobilhersteller verwendet das Tool Grafana, welches ebenso wie Kibana einen großen Funktionsumfang bietet (vgl. Kapitel 4.2.4). Den Rechenzentren des Automobilherstellers sind Cluster und Knoten verschiedener Ebenen untergeordnet. Insgesamt streckt sich die Hierarchie somit auf bis zu sechs Ebenen (gegenüber drei Ebenen bei der BA) und beinhaltet auf unterster Ebene ebenfalls eine Serveranzahl in der Größenordnung 10.000.

In der Grafana Oberfläche nutzt der Automobilhersteller eine Vielzahl an Diagrammen und zugehörigen Tabellen und Kennzahlen. Jedoch muss bei diesen analog zur BA zwischen den Ansichten und Parametern gewechselt werden, um Anomalien erkennen zu können. Insgesamt schöpft der Automobilhersteller das Potenzial von Grafana demnach nicht voll aus.

4.2 ITOA Tools

In den folgenden Unterkapiteln werden zwölf Tools beschrieben, die Auslastungsdaten von Rechenzentren oder einzelnen Servern visualisieren können bzw. die dafür genutzt oder deren Funktionen darauf übertragen werden können. Die Liste an Tools wurde im Rahmen der Literaturrecherche generiert und wird im Folgenden in alphabetischer Reihenfolge vorgestellt:

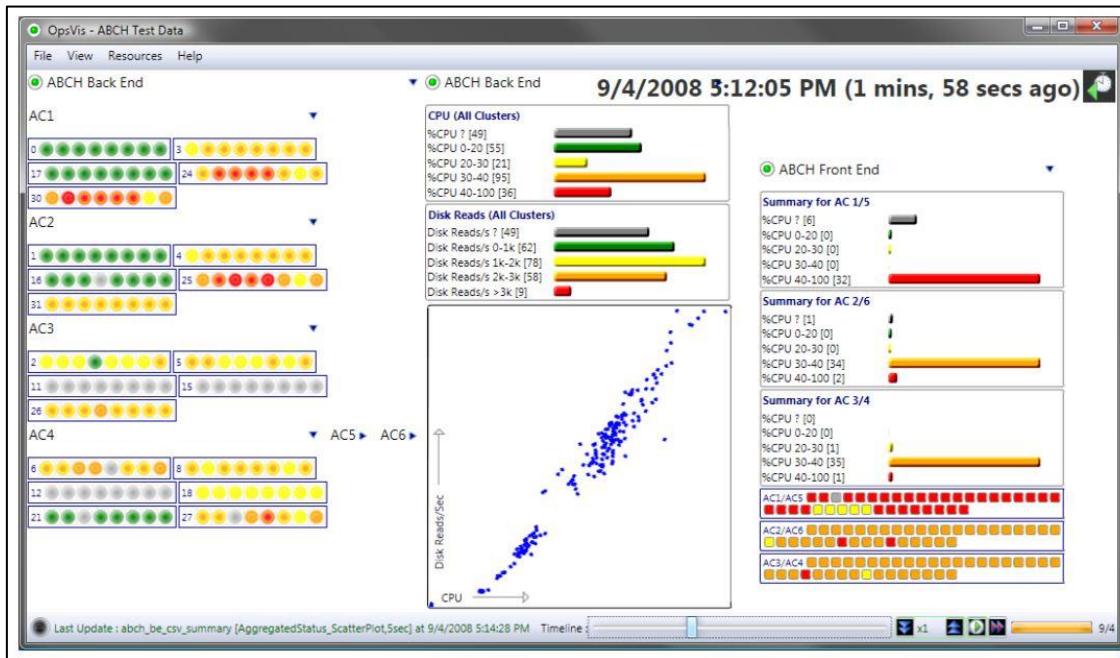
- | | |
|-------------|----------------------------|
| 1. ABCH | 7. Nagios |
| 2. Acrolinx | 8. Oracle IT Analytics |
| 3. ExtraHop | 9. Splunk |
| 4. Grafana | 10. TrueSight |
| 5. Kibana | 11. Windows Admin Center |
| 6. Munin | 12. Windows Capacity Check |

Jedes Unterkapitel beginnt mit einer kurzen Beschreibung des jeweiligen Tools und einem Screenshot der Oberfläche. Darauf folgen eine detailliertere Beschreibung des Funktionsumfanges und ein Fazit, inwiefern das jeweilige Tool in der Lage ist, eine große Serverlandschaft auf einen Blick darzustellen.

Sämtliche genutzte Informationen basieren auf den Beschreibungen und den Screenshots der jeweiligen Systeme. Da die Systeme nicht alle selbst getestet werden konnten, ist nicht auszuschließen, dass einzelne Systeme einen größeren Funktionsumfang besitzen, als im Folgenden dargestellt wird.

4.2.1 ABCH

Der Cloud Service ABCH (Address Book Clearing House) aus dem Jahr 2008 dient dazu, Adressbücher und Anwesenheitsinformationen von Nutzern des Windows Live Messengers zu speichern. Um die dafür nötigen Server zu überwachen, wurden relevante Metriken in einem Dashboard visualisiert [DHPW2008, TaBM2007].



[TaBM2007]

Abbildung 4: Screenshot ABCH

Überwacht wurden die CPU-Auslastung in verschiedenen Ansichten sowie die Speicherzugriffe pro Sekunde. Zur Darstellung wurden je nach Auslastung unterschiedlich gefärbte Kreise sowie Balkendiagramme und ein Punktdiagramm verwendet. Unten im Dashboard sieht man einen Datum-Slider sowie einen Zeitstempel für die aktuell gewählte Zeit (Abbildung 4).

Insgesamt ist ABCH stark veraltet, bietet einen sehr geringen Funktionsumfang und bietet keine Funktion, um eine Vielzahl an Servern übersichtlich darzustellen.

4.2.2 Acrolinx

Als Spin-off des Deutschen Forschungszentrums für Künstliche Intelligenz „liest“ Acrolinx den Content von Unternehmen und hilft diesen ansprechender, unterhaltender und wirkungsvoller zu gestalten. Dazu werden linguistische Merkmale verschiedener Sprachen analysiert. Ihr Spezialgebiet liegt in der Umsetzung von strategischen Kommunikationszielen [Acro2019a].



[Acro2019b]

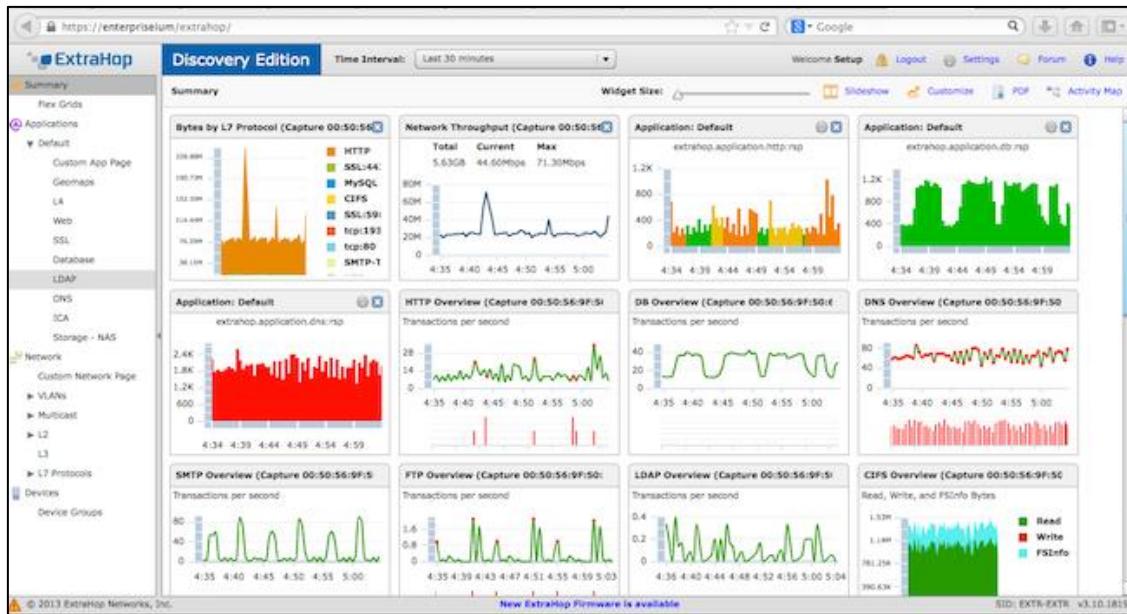
Abbildung 5: Screenshot Acrolinx

In einem Dashboard kann angezeigt werden, wie häufig und intensiv Acrolinx im Unternehmen genutzt wird. Mittels Linien- und Säulendiagrammen werden die in der obenstehenden Abbildung dargestellten Parameter visualisiert. Darüber hinaus bietet das Tool einen Slider, um einen gewünschten Zeitbereich auszuwählen (Abbildung 5) [Acro2019b].

Für die Systemkonzeption ist Acrolinx ungeeignet, da es nur Daten bezüglich der Nutzung des Tools selbst anzeigt. Es können jedoch die Art der Darstellung sowie der Slider als Input für die Konzeption verwendet werden.

4.2.3 ExtraHop

ExtraHop ist ein Unternehmen für Cyberanalytik welches Kunden bei der Erkennung und Klassifizierung von sämtlichen Assets im Unternehmen unterstützt. Dazu können in Echtzeit Verbindungen, Geräte und Benutzer sowie Abhängigkeiten zwischen diesen analysiert und visualisiert werden. Auf Basis von ML-Systemen können erweiterte Analysen durchgeführt und Anomalien erkannt werden [Extr2019].



[Extr2019]

Abbildung 6: Screenshot ExtraHop

Neben einer Vielzahl an Diagrammen (Abbildung 6) können auch dreidimensionale Darstellungen verwendet werden. Aus den zugehörigen Beschreibungen lässt sich jedoch ableiten, dass die dreidimensionalen Darstellungen nicht für die Visualisierung von Auslastungsdaten optimiert sind [Extr2019].

Insgesamt bietet ExtraHop umfassende Optionen zur Überwachung von Serverdaten. Das Tool besitzt jedoch kein Konzept, wie die einzelnen Darstellungen angeordnet und konfiguriert werden müssen, um eine große Serverlandschaft auf einen Blick darzustellen.

4.2.4 Grafana

Grafana ist eine Analyseplattform, die sich zum Ziel gesetzt hat, sämtliche Metriken aus beliebigen Quellen zusammen in einem Dashboard zu visualisieren [Graf2019].



[Graf2019]

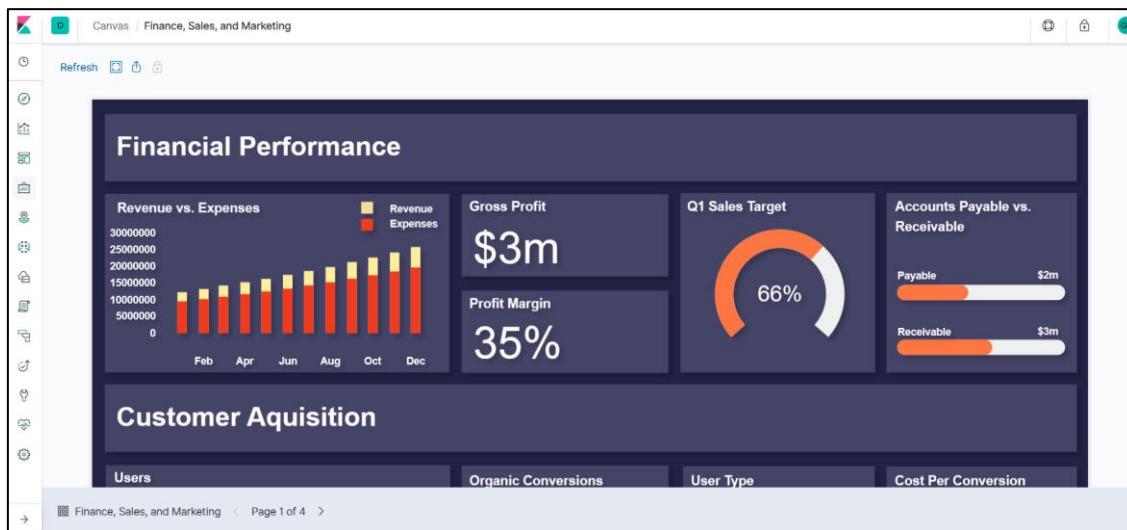
Abbildung 7: Screenshot Grafana

Grafana bietet eine Vielzahl an möglichen Darstellungsformen für die angebundenen Datenquellen. Das dargestellte Dashboard kann frei konfiguriert werden und bietet dem Benutzer ansprechende Interaktionsmöglichkeiten (Abbildung 7). Darüber hinaus ist es auch möglich Alerts zu erhalten, wenn selbst festgelegte Grenzwerte über- oder unterschritten werden [Graf2019]. Grafana unterstützt mit der Erweiterung Plotly auch eigene dreidimensionale Darstellungen [GrPl2019].

Insgesamt ist Grafana sehr nutzerfreundlich und bietet einen großen Funktionsumfang, der zusätzlich noch durch Erweiterungen ergänzt werden kann. Grundsätzlich ist Grafana ein Open Source Programm, bietet jedoch auch eine kostenpflichtige Enterprise Version an [Graf2019]. Aufgrund der flexiblen Einsetzbarkeit von Grafana existiert keine Vorlage, wie das Dashboard konfiguriert werden müsste, um die gesamte Serverlandschaft eines Unternehmens übersichtlich auf einen Blick anzuzeigen.

4.2.5 Kibana

Kibana ist eine Analyseplattform, welche auf der Suchmaschine Elasticsearch aufbaut. Kibana ist sehr nutzerfreundlich und unterstützt alle klassischen Visualisierungen (Abbildung 8). Darüber hinaus können mit Hilfe der Vega-Grammatik eigene Visualisierungen erstellt werden [Elas2019, Kiba2019, Vega2019]. Beispielhafte Dashboards und Visualisierungen lassen sich in der Kibana-Demo finden [KiDe2019].



[Kiba2019]

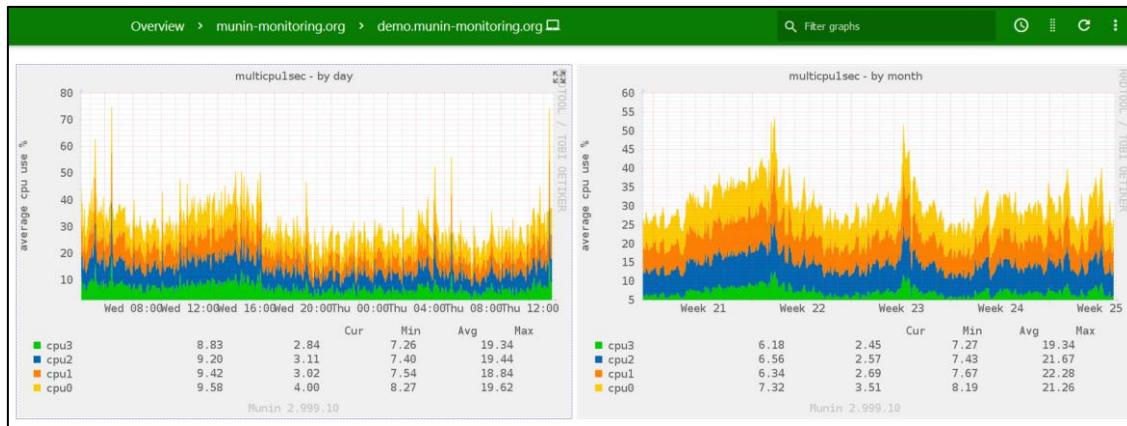
Abbildung 8: Screenshot Kibana

In Kibana können grundsätzlich alle Metriken visualisiert werden, welche in der Elasticsearch-Datenbank vorhanden sind. Über entsprechende Plugins können auch dreidimensionale Diagramme dargestellt werden [Rusu2017]. Ein ML-Plugin ermöglicht es zudem Anomalien zu finden und Prognosen zu erstellen. Einige der angebotenen Plugins stehen kostenlos zur Verfügung. Für den vollen Funktionsumfang inklusive Onlinehosting fallen jedoch monatliche Gebühren an [Kiba2019].

Insgesamt ist Kibana ein mächtiges und nutzerfreundliches System, das durch eine Vielzahl an Plugins nahezu alle Anforderungen erfüllen kann. Dabei wird man jedoch durch die Lizenzkosten sowie durch die konkreten Möglichkeiten der Plugins beschränkt. Darüber hinaus kann Kibana lediglich als Basis für das zu konzipierende System dienen, die konkrete Konzeption von Oberfläche und Funktionalität wäre auch bei der Nutzung von Kibana nötig [Kiba2019].

4.2.6 Munin

Munin ist ein Tool zur Überwachung von Netzwerk-Ressourcen und läuft auf den meisten gängigen Linux-Distributionen. Das Tool stellt eine Vielzahl an zweidimensionalen Diagrammen bereit, welche sich jeweils auf verschiedene Zeiteinheiten skalieren lassen. Über Plugins können zusätzliche Darstellungen verwendet werden [Muni2019].



[Muni2019]

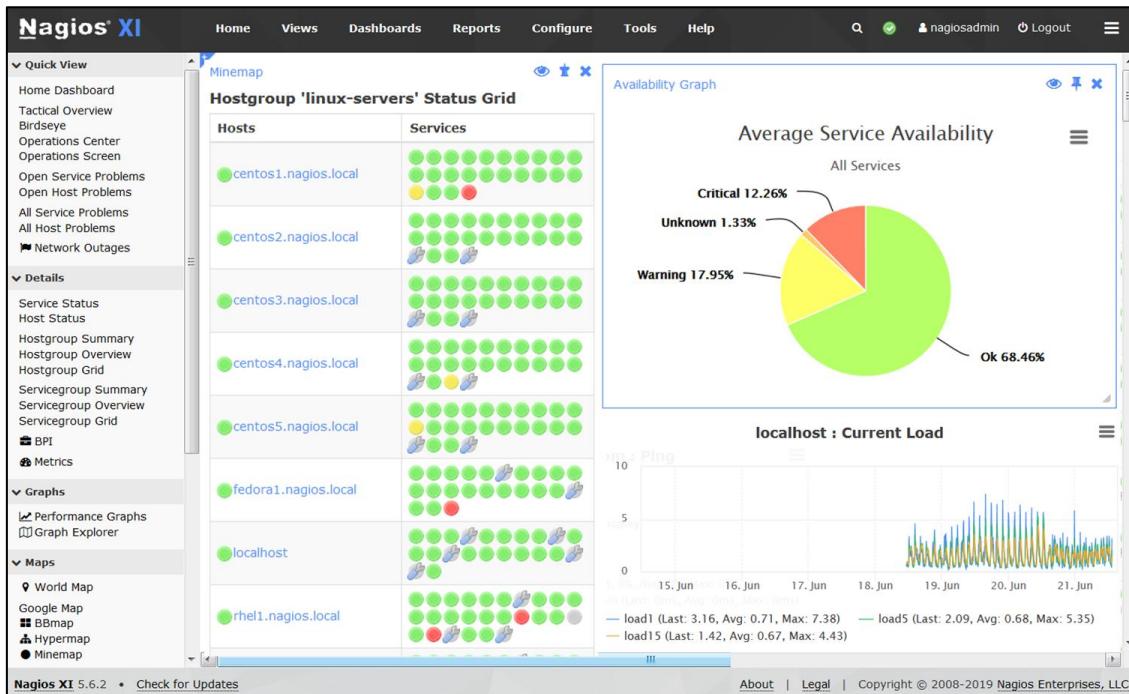
Abbildung 9: Screenshot Munin

Munin stellt mittels Linien- und Flächendiagrammen aktuelle, minimale, maximale und Durchschnittswerte verschiedener Metriken dar. Dargestellte Metriken sind zum Beispiel CPU-Auslastung, Lese- und Schreibzugriffe pro Sekunde, Dateisystemnutzung und Datenträgerlatenz. Alle Diagramme können auf Basis von Stunden, Tagen, Wochen, Monaten und Jahren angezeigt werden. Zusätzlich ermöglicht eine Zoomfunktion den gewählten Zeitbereich zu verfeinern. Die Diagramme werden in einem wählbaren Zeitintervall automatisch aktualisiert. Ein separates Menü listet Probleme wie Warnungen und Fehler auf [Muni2019] (Abbildung 9).

Insgesamt zeigt das Tool zwar sehr viele Diagramme an, unter diesen finden sich jedoch nur zwei verschiedene Diagrammtypen. Bei Bedarf können eigene Metriken ergänzt und visualisiert sowie Plugins für zusätzliche Funktionen installiert werden. Eine dreidimensionale Darstellung ist in der Munin-Plugin-Galerie jedoch nicht verfügbar. Darüber hinaus müssen die meisten Einstellungen im Programmcode vorgenommen werden, sodass sie nicht für jeden Benutzer einfach zugänglich sind [Muni2019].

4.2.7 Nagios

Nagios ist ein System zur Überwachung von komplexen IT-Infrastrukturen und darauf spezialisiert, Probleme zu identifizieren und zu lösen, bevor sie sich auf kritische Geschäftsprozesse auswirken [Nagi2019].



[Nagi2019]

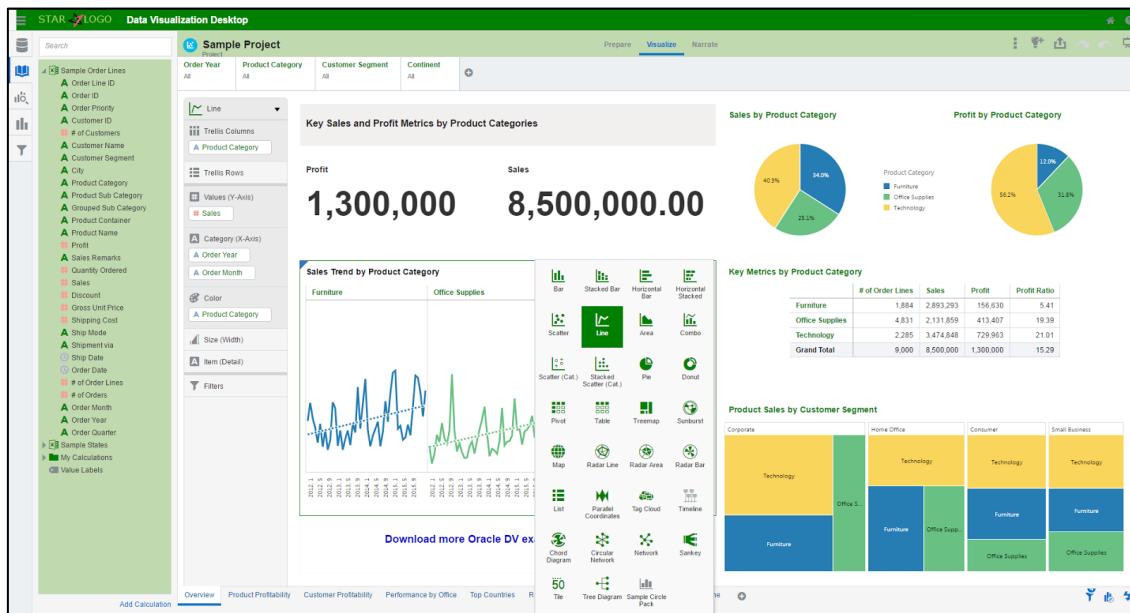
Abbildung 10: Screenshot Nagios

Die Software stellt Auslastungsdaten in einem Dashboard dar (Abbildung 10) und bietet darüber hinaus spezialisierte Ansichten und ein automatisiertes Alerting bei kritischen Werten oder Zuständen. Nagios wirbt mit einer schnellen Erkennung von Netzwerk- und Serverausfällen sowie dem Ausfall von Diensten und Prozessen [Nagi2019].

Zur Visualisierung der Serverdaten werden klassische Diagrammtypen und Darstellungsformen angeboten. Von diesen ist jedoch keine in der Lage, eine komplexe Serverlandschaft übersichtlich auf einen Blick darzustellen. Darüber hinaus ist es nicht möglich eigene dreidimensionale Diagramme einzubinden.

4.2.8 Oracle IT Analytics

Der Oracle IT Analytics Cloud Service hat das Ziel, Einsicht in Performance, Verfügbarkeit und Kapazität von Anwendungen zu geben. Des Weiteren soll es Analysten, IT-Führungskräften und Administratoren ermöglichen, kritische Entscheidungen zu treffen [Orac2019].



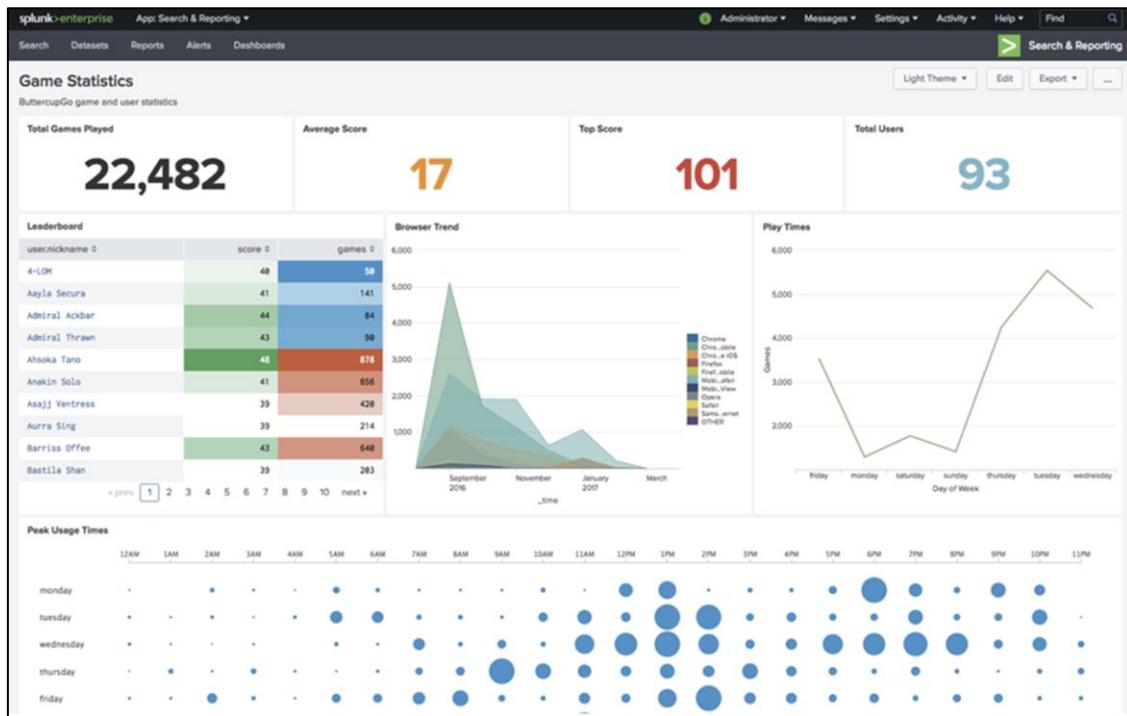
[Orac2019]

Abbildung 11: Screenshot Oracle IT Analytics

Oracle IT Analytics wirbt damit, Auslastungen zu optimieren, indem verschiedene Parameter gleichzeitig analysiert werden. Zusätzlich ermöglicht der Service, Prognosen zu erstellen und den Benutzer bei kritischen Zuständen automatisch zu alarmieren. Diese Funktionen können in einem selbst konfigurierbaren Dashboard visualisiert werden (Abbildung 11). Auf der Webseite von Oracle wurden nur wenige Screenshots der Oberfläche bereitgestellt [Orac2019]. Der oben dargestellte Screenshot basiert auf übersichtlich angeordneten, klassischen Diagrammen. Aus den verfügbaren Servicebeschreibungen geht nicht hervor, dass Oracle IT Analytics eine kompakte Darstellung von einer Vielzahl an Servern unterstützt.

4.2.9 Splunk

Splunk ist darauf spezialisiert, Maschinendaten für Benutzer zugänglich zu machen. Dazu können die Daten in Echtzeit und mit Unterstützung von ML analysiert und in Dashboards angezeigt werden [Splu2019].



[Splu2019]

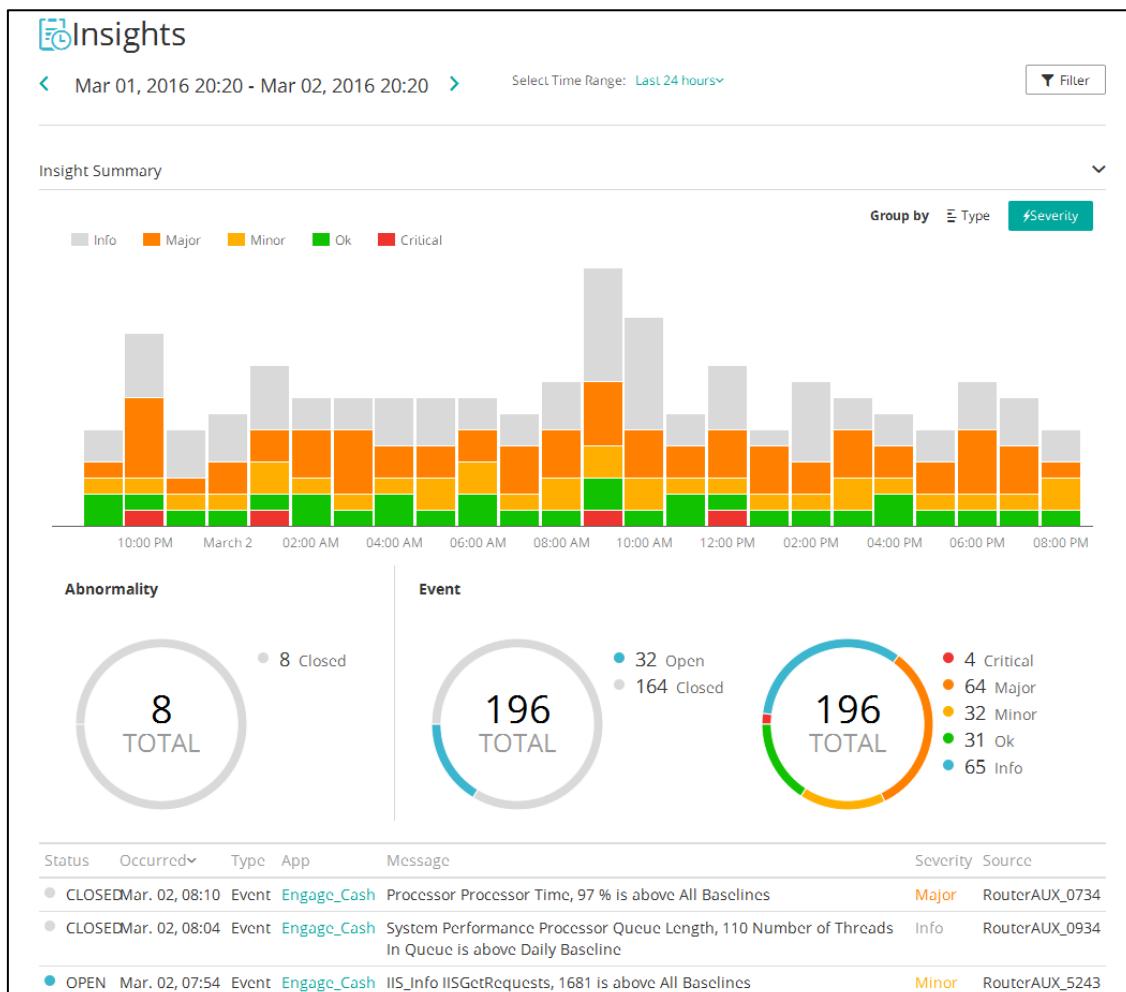
Abbildung 12: Screenshot Splunk

Splunk bietet eine große Bandbreite an Darstellungsoptionen (Abbildung 12) und ermöglicht die Erstellung von eigenen und dreidimensionalen Diagrammen. Diese können mit D3.js erstellt werden, wodurch dem Benutzer umfassende Möglichkeiten geboten werden [Splu2019].

Insgesamt ist Splunk ein mächtiges Tool, das durch die Anbindung von D3.js auch komplexe und dreidimensionale Diagramme darstellen kann. Splunk fehlt lediglich ein Konzept zum Aufbau der Oberfläche und zur Veranschaulichung der Diagramme, um die Serverlandschaft auf einen Blick zu zeigen.

4.2.10 TrueSight

TrueSight Capacity Optimization vom Softwarehersteller BMC „stimmt die IT-Ressourcen auf die Serviceanforderungen ab und ermöglicht so eine Optimierung der Ressourcennutzung und Kosten“ [True2019].



[True2019]

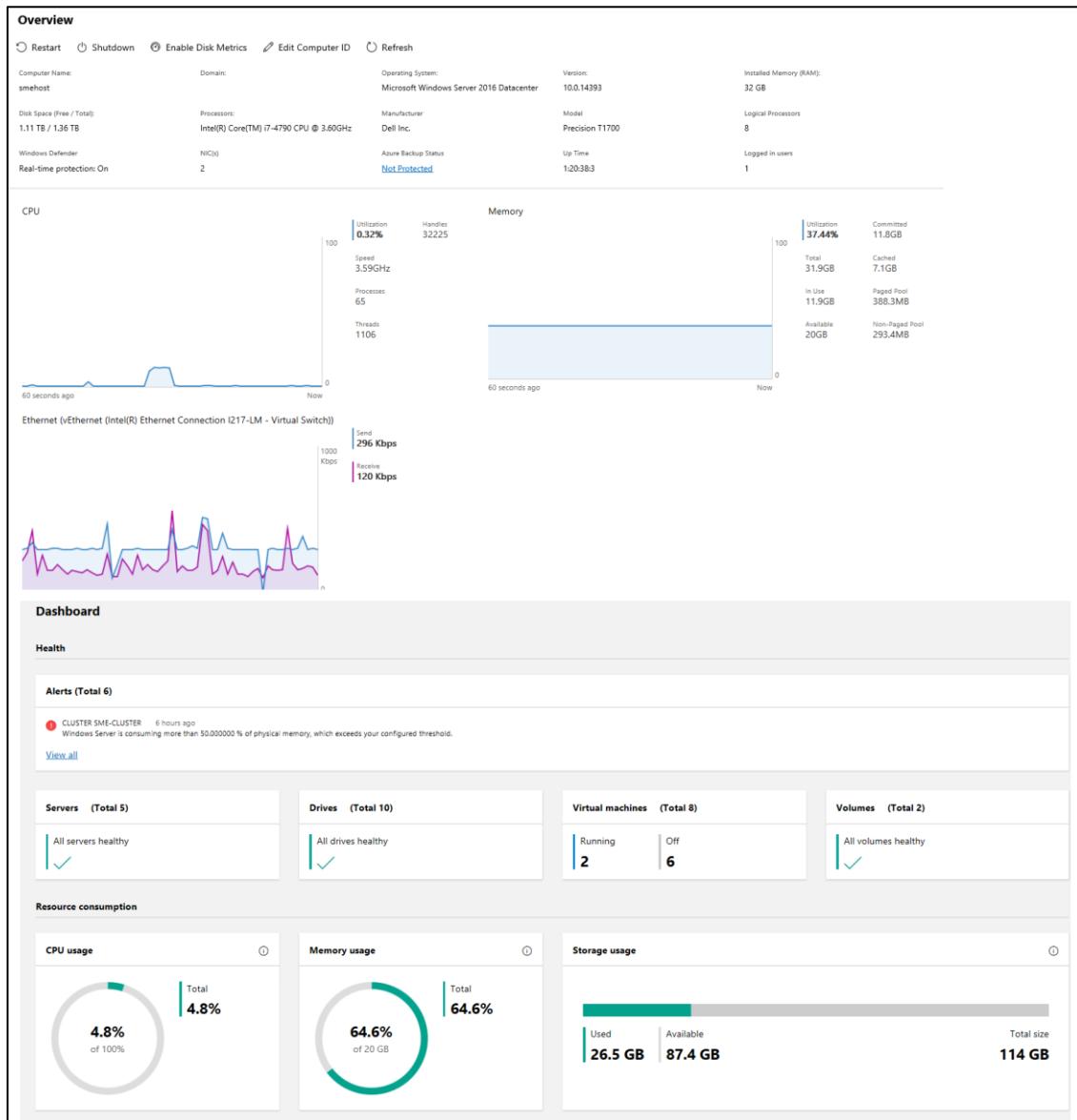
Abbildung 13: Screenshot TrueSight

TrueSight ermöglicht es, Auswirkungen von Ressourcenveränderungen zu simulieren und Prognosen anzuzeigen. Darüber hinaus können Berichte erstellt und mittels ML Anomalien gefunden werden [True2019] (Abbildung 13).

Insgesamt wird TrueSight als ganzheitliche Lösung im Multi-Cloud Management beworben. Dadurch eignet sich das Tool nicht für die Analyse von Auslastungsdaten von lokalen Rechenzentren.

4.2.11 Windows Admin Center

Das Windows Admin Center als eine Weiterentwicklung der Windows Server-Verwaltungstools ist eine „browserbasierte App zum Verwalten von Servern, Clustern, hyperkonvergenten Infrastrukturen und Windows 10-PCs“ [Micr2019]



[Micr2019]

Abbildung 14: Screenshot Windows Admin Center

In einem Dashboard (Abbildung 14) zeigt das Windows Admin Center Leistungsindikatoren zu Prozessoren, Arbeitsspeicher, Cache, Threads und Prozessen an. Insgesamt ist das System auf die Überwachung von Windows-Servern ausgelegt. Somit lassen sich für die Konzeption nur Aufbau und Art der Darstellungen übernehmen.

4.2.12 Windows Capacity Check

Das kostenlose Tool Windows Capacity Check kann kleinere IT-Umgebungen überwachen und dazu Daten zu CPU-Auslastung, RAM und Netzwerk in einem Dashboard darstellen (Abbildung 15) [Somm2015].



[Somm2015]

Abbildung 15: Screenshot Windows Capacity Check

Das Tool ermöglicht es, dem Benutzer Warnungen beim Überschreiten definierter Grenzwerte festzulegen. Diese Funktion kann für die Konzeption adaptiert werden. Insgesamt ist Windows Capacity Check stark veraltet und nicht in der Lage größere Serverlandschaften darzustellen.

5 Anforderungen an die Neukonzeption

In diesem Kapitel werden die Anforderungen an das zu konzipierende System gesammelt und zu einer Anforderungsliste aggregiert.

5.1 Anforderungen aus ITOA

Die mayato GmbH, die Unternehmen im Bereich ‚End-to-end Business Analytics‘ unterstützt, hat in einem Whitepaper Anforderungen aufgelistet. Diese müssen von ITOA-Systemen erfüllt werden [Maya2016]. Die Anforderungen sind in vier Kategorien unterteilt und sollen als Anforderungen für das im Rahmen dieser Arbeit zu konzipierende System herangezogen werden (Tabelle 5). Die Nummerierung der Anforderungen entspricht der in der konsolidierten Anforderungsliste (Kapitel 5.4).

Nr.	Anforderung
1. Kernanforderungen	
1.5	Einfache Handhabung
1.6	Kosten
1.7	Unterstütze Datenquellen
1.8	Einfach Anpassbarkeit von Dashboard und Reports
1.9	Qualität der Visualisierungen
2. Funktionalität	
2.1	Das System muss eine intuitive, moderne und gut strukturierte Benutzeroberfläche besitzen
2.2	Das Erstellen von Dashboards muss realisierbar sein
2.3	Ein Alerting beim Überschreiten von Grenzwerten muss eingerichtet werden können
2.4	Die Auswertung von historischen Daten muss möglich sein
2.5	Das Durchsuchen von Logdaten muss durchführbar sein
2.6	Das System muss Analytics Funktionen wie bspw. Machine Learning unterstützen oder Schnittstellen zu ‚R‘ oder Python anbieten
4. Sicherheit (siehe Anhang)	
5. Architektur (siehe Anhang)	

[IDC2015]

Tabelle 5: Anforderungen aus ITOA

Kernanforderungen sind Anforderungen mit höchster Priorität. Die funktionalen Anforderungen beschreiben, welche Funktionen das System erfüllen muss. Die Anforderungen Sicherheit und Architektur sind Teil der technischen Umsetzung und somit nicht Bestandteil dieser Arbeit. Eine vollständige Auflistung der letzten beiden Kategorien befindet sich im Anhang (Tabelle 20).

5.2 Anforderungen der Benutzer

Durch das agile Vorgehensmodell konnten die Benutzer in jeder Phase Feedback geben und eigene Anforderungen ergänzen. Diese Anforderungen lassen sich in die Gruppen ‚Kernanforderungen‘, ‚Usability‘ und ‚Funktionalität‘ aufteilen und werden im Folgenden näher erläutert.

Kernanforderungen (Anforderungen mit höchster Priorität)

Nr.	Anforderung
1.1	Fehler und Anomalien bemerken
1.1.1	Das System muss dem Benutzer anzeigen, wenn ein Fehler oder eine Anomalie vorliegt
1.1.2	Das System muss den Benutzer visuell unterstützen, Fehler und Anomalien zu bemerken
1.2	Fehler und Anomalien klassifizieren
1.2.1	Das System muss dem Benutzer anzeigen, welche Art von Fehler oder Anomalie vorliegt
1.2.2	Das System muss den Benutzer visuell unterstützen zu klassifizieren, welche Art von Fehler oder Anomalie vorliegt
1.3	Fehler und Anomalien lokalisieren
1.3.1	Das System muss dem Benutzer anzeigen, wo ein Fehler oder eine Anomalie vorliegt
1.3.2	Das System muss den Benutzer visuell unterstützen zu lokalisieren, wo ein Fehler oder eine Anomalie vorliegt
1.4	Alle relevanten Informationen in einer Ansicht bereitstellen

Tabelle 6: Anforderungen der Benutzer – Kernanforderungen

Die Komplexität der Serverumgebung und die hohe Kritikalität der Dienste erfordern ein umfassendes und effizientes Monitoring. Daher muss das System in der Lage sein, Fehler und Anomalien frühzeitig zu bemerken sowie diese zu klassifizieren und zu lokalisieren [CSB+2019]. Diese drei Anforderungen werden in Teilanforderungen untergliedert, die in der oben angeführten Tabelle dargestellt sind. Darüber hinaus muss das System alle für das Monitoring nötigen Informationen in einer Ansicht bereitstellen, ohne dass zwischen verschiedenen Ansichten und Tools gewechselt werden muss (Tabelle 6).

Usability

Die Benutzer haben hohe Anforderungen an die Usability des Systems. Dazu wurden unter anderem folgende Punkte genannt: Einheitliches Design, ausreichende Einstellungsmöglichkeiten, die Möglichkeit zur Verwendung von Shortcuts sowie die Möglichkeit Aktionen rückgängig zu machen. Da alle von den Benutzern genannten Usability-Anforderungen Teil der in Tabelle 3 genannten Kriterien sind, sollen diese als zu erfüllende Anforderungen aufgenommen werden.

Funktionalität

Stichwort	Beschreibung
,Go live'-Schalter	Möglichkeit umzuschalten, ob Daten live oder auf Basis von Vergangenheitsdaten angezeigt werden.
Datum-Slider für 3D-Diagramm	Zusätzlich zur Auswahl von Datum und Uhrzeit über das Menü soll ein Slider unterhalb vom 3D-Diagramm angezeigt werden. Es soll einstellbar sein, ob sich das Diagramm bei Änderung des Sliders live aktualisiert oder erst nachdem sich der Benutzer für einen Wert entschieden hat. Dies soll Ruckeln bei zu großen Datenmengen verhindern.
Zeitstempel mit Wochentag	Damit niedrige Auslastungswerte an Wochenenden nicht fälschlicherweise als Fehler angesehen werden, soll der Wochentag im Zeitstempel mit angezeigt werden.
Erstellung eines Reports	Um Prognosen und Fehler an höhere Hierarchieebenen weiterleiten zu können, wird die automatische Erstellung von PDF-Reports gewünscht.
Metriken anpassbar machen	Die Konzeption sah anfangs eine feste Liste von zur Auswahl stehenden Metriken vor. Um mit verschiedenen Datenquellen und Formaten sowie zusätzlichen Metriken umgehen zu können, bestand der Wunsch, dass auch eigene Metriken hinzugefügt werden können.
Parameter ,Temperatur' entfernen	Die Konzeption sah anfangs als Parameter verschiedene Temperaturwerte vor. Die Hardware-Wartung in Rechenzentren wird jedoch meistens nicht von denjenigen durchgeführt, die die weiteren Metriken überwachen. Die Temperaturdaten stehen daher in der Regel nur den Mitarbeitern der Hardware-Wartung zur Verfügung.

Tabelle 7: Anforderungen der Benutzer – Funktionalität

Bezüglich Funktionalität wurden Anforderungen auf Basis von Mockups und dem Prototypen gestellt. In der obenstehenden Tabelle werden diese Anforderungen jeweils kurz genannt und daneben ausführlich beschrieben (Tabelle 7).

5.3 Parameter

Die Diagramme im Dashboard sollen standardmäßig die CPU-Auslastung anzeigen. Auf Basis existierender Tools und Anforderungen der Benutzer wurden weitere Metriken gesammelt, die anstelle dessen oder auch zusätzlich zu der CPU-Auslastung angezeigt werden können (Tabelle 8).

Parameter	Zahlenformat	Minimum	Maximum	Standard-skala
CPU-Auslastung	0,00 %	0,00	100,00	Linear
Speicherzugriffe	0 pro Sekunde	0	max	Linear
Arbeitsspeicher-auslastung	0,00 %	0,00	100,00	Linear
Netzwerk-auslastung	0,00 %	0,00	100,00	Linear
Anzahl Prozesse	0	0	max	Logarithmisch
Antwortzeit	0,00 Millisekunden	0,00	max	Logarithmisch
Anzahl Anfragen	0 pro Sekunde	0	max	Linear

Tabelle 8: Parameter

Im Folgenden werden die einzelnen Tabellenspalten näher erläutert.

Parameter

Diese Spalte listet alle zur Auswahl stehenden Metriken auf. Im System können darüber hinaus eigene Metriken auf Basis der verwendeten Datenquelle konfiguriert werden.

Zahlenformat

Diese Spalte gibt an, wie die Werte der Metriken dargestellt werden sollen. Die Angabe 0,00 % bei der CPU-Auslastung bedeutet beispielsweise, dass der Wert auf zwei Nachkommastellen und in Prozent angegeben wird.

Minimum und Maximum

Die Minimum- und Maximum-Werte geben die Grenzen der Diagramm-Achsen an bzw. legen bei aktiviertem Farbcode zusammen mit der Standardskala fest, welche Farbe jeweils angewendet wird.

Einige der Grenzwerte sind fest vorgegeben, da keine Werte außerhalb des definierten Bereichs vorkommen können (z. B. CPU-Auslastung zwischen 0 % und 100 %). Andere Werte sollen ihr Maximum durch den höchsten Wert im gewählten Zeitintervall erhalten, wodurch auch extreme Werte dargestellt werden können (z. B. wenn die Antwortzeit in einem Zeitintervall ungewöhnlich groß ist).

Standardskala

Entlang der Diagrammachsen sind die Werte entsprechend einer ‚Standardskala‘ verteilt. Diese Skala legt auch fest, wie sich die Werte bei aktivem Farbcode in die voreingestellten fünf Farben gliedern. So wechselt beispielsweise bei der CPU-Auslastung die Farbe linear alle 20%, während bei der Anzahl an Threads eine logarithmische Skala verwendet wird. Die Art der Standardskala kann individuell für alle Metriken angepasst werden. Zur Auswahl stehen neben linear auch verschiedene logarithmische Abstufungen sowie die Möglichkeit eigene Berechnungen zu hinterlegen.

5.4 Anforderungsliste

Die untenstehende Tabelle bündelt alle Anforderungen der vorherigen Kapitel und dient als Basis für die Konzeption des neuen Systems. Die Anforderungen sind in fünf Kategorien gruppiert und zur leichteren Evaluation nummeriert. Neben jeder Anforderung steht, aus welchem Kapitel sie jeweils stammt (Tabelle 9).

Nr.	Anforderung	Quelle der Anforderung
1. Kernanforderungen		
1.1	Fehler und Anomalien bemerken	
1.1.1	Das System muss dem Benutzer anzeigen, wenn ein Fehler oder eine Anomalie vorliegt	Benutzer (5.2)
1.1.2.	Das System muss den Benutzer visuell unterstützen, Fehler und Anomalien zu bemerken	Benutzer (5.2)
1.2	Fehler und Anomalien klassifizieren	
1.2.1	Das System muss dem Benutzer anzeigen, welche Art von Fehler oder Anomalie vorliegt	Benutzer (5.2)
1.2.2	Das System muss den Benutzer visuell unterstützen zu klassifizieren, welche Art von Fehler oder Anomalie vorliegt	Benutzer (5.2)
1.3	Fehler und Anomalien lokalisieren	
1.3.1	Das System muss dem Benutzer anzeigen, wo ein Fehler oder eine Anomalie vorliegt	Benutzer (5.2)
1.3.2	Das System muss den Benutzer visuell unterstützen zu lokalisieren, wo ein Fehler oder eine Anomalie vorliegt	Benutzer (5.2)
1.4	Alle relevanten Informationen in einer Ansicht bereitstellen	Benutzer (5.2)
1.5	Einfache Handhabung	ITOA (5.1)
1.6	Kosten	ITOA (5.1)
1.7	Unterstütze Datenquellen	ITOA (5.1)
1.8	Einfach Anpassbarkeit von Dashboard und Reports	ITOA (5.1)
1.9	Qualität der Visualisierungen	ITOA (5.1)
2. Funktionalität		
2.1	Das System muss eine intuitive, moderne und gut strukturierte Benutzeroberfläche besitzen	ITOA (5.1)
2.2	Das Erstellen von Dashboards muss realisierbar sein	ITOA (5.1)
2.3	Ein Alerting beim Überschreiten von Grenzwerten muss eingerichtet werden können	ITOA (5.1)
2.4	Die Auswertung von historischen Daten muss möglich sein	ITOA (5.1)
2.5	Das Durchsuchen von Logdaten muss durchführbar sein	ITOA (5.1)
2.6	Das System muss Analytics Funktionen wie bspw. Machine Learning unterstützen oder Schnittstellen zu ‚R‘ oder Python anbieten	ITOA (5.1)
2.7	Unterstützung der in Tabelle 8 aufgelisteten sowie Möglichkeit zur Verwendung eigener Parameter	Parameter (5.3)
2.8	„Go live“-Schalter	Benutzer (5.2)
2.9	Datum-Slider für 3D-Diagramm	Benutzer (5.2)
2.10	Zeitstempel mit Wochentag	Benutzer (5.2)
2.11	Erstellung eines Reports	Benutzer (5.2)

Nr.	Anforderung	Quelle der Anforderung
3.	Usability	
3.1	Gib Benutzern die Kontrolle	
3.1.1	Nutze Modi mit Bedacht (gleiche Buttons = gleiche Aktionen)	Usability (3.3.2)
3.1.2	Ermögliche die Bedienung jeweils mit Tastatur und Maus	Usability (3.3.2)
3.1.3	Ermögliche Benutzern den Fokus zu ändern	Usability (3.3.2)
3.1.4	Zeige beschreibende Nachrichten und Texte an	Usability (3.3.2)
3.1.5	Ermögliche sofortige und reversible Aktionen	Usability (3.3.2)
3.1.6	Sinnvolle Navigation mit Abbruchmöglichkeiten	Usability (3.3.2)
3.1.7	Berücksichtige Benutzer verschiedener Qualifikationsstufen	Usability (3.3.2)
3.1.8	Machen Sie die Benutzeroberfläche transparent (der Benutzer sollte sich jederzeit voll auf seine Aufgabe fokussieren können)	Usability (3.3.2)
3.1.9	Ermögliche Benutzern das Anpassen der Benutzeroberfläche	Usability (3.3.2)
3.1.10	Ermögliche Benutzern das direkte Bearbeiten von Objekten	Usability (3.3.2)
3.2	Reduziere den Gedächtnisaufwand der Benutzer	
3.2.1	Entlaste das Kurzzeitgedächtnis	Usability (3.3.2)
3.2.2	Unterstütze den Benutzer dabei Elemente wiederzuerkennen (z. B. durch Tooltips oder kontextabhängige Hinweise)	Usability (3.3.2)
3.2.3	Stelle visuelle Hinweise bereit	Usability (3.3.2)
3.2.4	Ermögliche Standardeinstellungen sowie Aktionen rückgängig zu machen und zu wiederholen	Usability (3.3.2)
3.2.5	Stelle Shortcuts bereit	Usability (3.3.2)
3.2.6	Strebe eine objektorientierte Syntax an	Usability (3.3.2)
3.2.7	Verwende reale Metaphern	Usability (3.3.2)
3.2.8	Nutze progressive Offenlegung von Funktionen (zeige dem Benutzer Funktionen nur dann, wenn er sie benötigt)	Usability (3.3.2)
3.2.9	Strebe visuelle Klarheit an	Usability (3.3.2)
3.3	Strebe Konsistenz an	
3.3.1	Erhalte den Kontext der Aufgaben der Benutzer (Kontinuität)	Usability (3.3.2)
3.3.2	Stelle Konsistenz zu anderen Systemen her (ähnlicher Aufbau von Menüs und gleiche Funktionsweise von bekannten Buttons)	Usability (3.3.2)
3.3.3	Gleiche Aktionen sollten zu gleichen Ergebnissen führen	Usability (3.3.2)
3.3.4	Strebe ein optisch ansprechendes Design an	Usability (3.3.2)
3.3.5	Ermutige den Benutzer Funktionen auszuprobieren	Usability (3.3.2)
4.	Sicherheit (siehe Anhang)	
5.	Architektur (siehe Anhang)	

Tabelle 9: Anforderungsliste

6 Zwischenfazit

Nach der Erfassung des Ist-Zustandes und der Anforderungen an die Neukonzeption soll die Analysephase des Vorgehensmodells (Phase 1) in diesem Kapitel mit einem Zwischenfazit abgeschlossen werden. Dazu wird begründet, warum eine Neukonzeption notwendig ist und warum diese eine dreidimensionale Darstellung enthalten sollte. Im Anschluss wird das System gestaltet (Phase 2, Kapitel 7), in Form eines Prototypen konstruiert (Phase 3, Kapitel 8) und evaluiert (Phase 4, Kapitel 9).

6.1 Notwendigkeit einer Neukonzeption

Die in Kapitel 4.2 untersuchten Tools lassen sich in zwei Gruppen unterteilen. Die meisten Tools sind auf die Visualisierung von Auslastungsdaten spezialisiert und bieten dazu eine vorgegebene Oberfläche. Diese Tools ermöglichen es jedoch nicht, die im vorherigen Kapitel genannten Anforderungen abzudecken. Konkret ermöglichen diese Tools nicht, die gesamte Serverlandschaft eines Unternehmens auf einen Blick übersichtlich darzustellen. Die zweite Gruppe an Tools ist nicht auf die Visualisierung von Auslastungsdaten spezialisiert und bietet daher keine vorgegebene Oberfläche dafür an. Diese Tools ermöglichen es jedoch die Oberfläche nach eigenem Bedarf anzupassen und unterstützen auch die Integration von dreidimensionalen Diagrammen.

Gruppe 1: Vorgegebene Oberfläche		Gruppe 2: Anpassbare Oberfläche
• ABCH	• Oracle IT Analytics	• ExtraHop
• Acrolinx	• TrueSight	• Grafana
• Munin	• Windows Admin Center	• Kibana
• Nagios	• Windows Capacity Check	• Splunk

Tabelle 10: Toolübersicht (gruppiert)

Um Rechenzentren überwachen zu können, wird ein System benötigt, welches die gesamte Serverlandschaft auf einen Blick darstellen kann und eine definierte Oberfläche besitzt. Um diese Anforderungen umzusetzen, müssen die Tools der Gruppe 1 um ein dreidimensionales Diagramm ergänzt werden. Da diese Tools jedoch keine dreidimensionalen Diagramme unterstützen, sind sie nicht in der Lage, die geforderten Anforderungen zu erfüllen. Die Tools der Gruppe 2 unterstützen dreidimensionale Diagramme. Ihnen fehlt lediglich eine definierte Oberfläche. Daraus resultiert die Notwendigkeit, ein System zu konzipieren, welches die geforderten Anforderungen erfüllt. Ob dieses Konzept in Kombination mit den Tools der zweiten Gruppe verwendet oder von Grund auf neu programmiert werden sollte, wird in Kapitel 11.1 diskutiert.

6.2 Vorteile einer dreidimensionalen Visualisierung

Eine dreidimensionale Visualisierung ermöglicht es, alle Server, Cluster und Rechenzentren eines Unternehmens auf einen Blick darzustellen. Dabei kann der Benutzer entscheiden, ob zwei gleiche oder zwei unterschiedliche Parameter abgebildet werden sollen. Dies kann dadurch erreicht werden, indem die Höhe der 3D-Balken und ein Farbcodet genutzt werden. Beim Einsatz von zwei gleichen Parametern wird die Erkennung von Auffälligkeiten verbessert. Werden zwei unterschiedliche Parameter verwendet, erhöht sich der Informationsgehalt. Darüber hinaus ermöglicht es die dreidimensionale Visualisierung Cluster und Rechenzentren räumlich gruppiert anzurufen. Dadurch können Zusammenhänge zwischen diesen deutlich besser erkannt werden. Unterstützt wird der Benutzer dabei, indem er die 3D-Ansicht beliebig zoomen und drehen kann. Eine Evaluation mit den Benutzern der Bundesagentur für Arbeit hat ergeben, dass die meisten von ihnen eine dreidimensionale Darstellung gegenüber dem bisherigen Tool vorziehen [CSB+2019].

7 Konzeption des neuen Systems

In diesem Kapitel werden alle Elemente des neuen Systems anhand von Mockups erläutert. Dazu wird zunächst der allgemeine Aufbau der Oberfläche beschrieben. Anschließend werden die einzelnen Ansichten, Panels und weiteren Elemente im Detail erläutert. Abschließend werden besondere Funktionalitäten beschrieben und im Styleguide werden grundsätzliche Designvorgaben und Gestaltungshinweise gegeben. Die Oberfläche ist an der von Kibana orientiert.

7.1 Allgemeiner Aufbau

Das System besteht aus den vier Bereichen ‚Schnelleinstellungen‘, ‚Navigationsleiste‘, ‚Hauptansicht‘ und ‚Detaileinstellungen‘ (Abbildung 16).

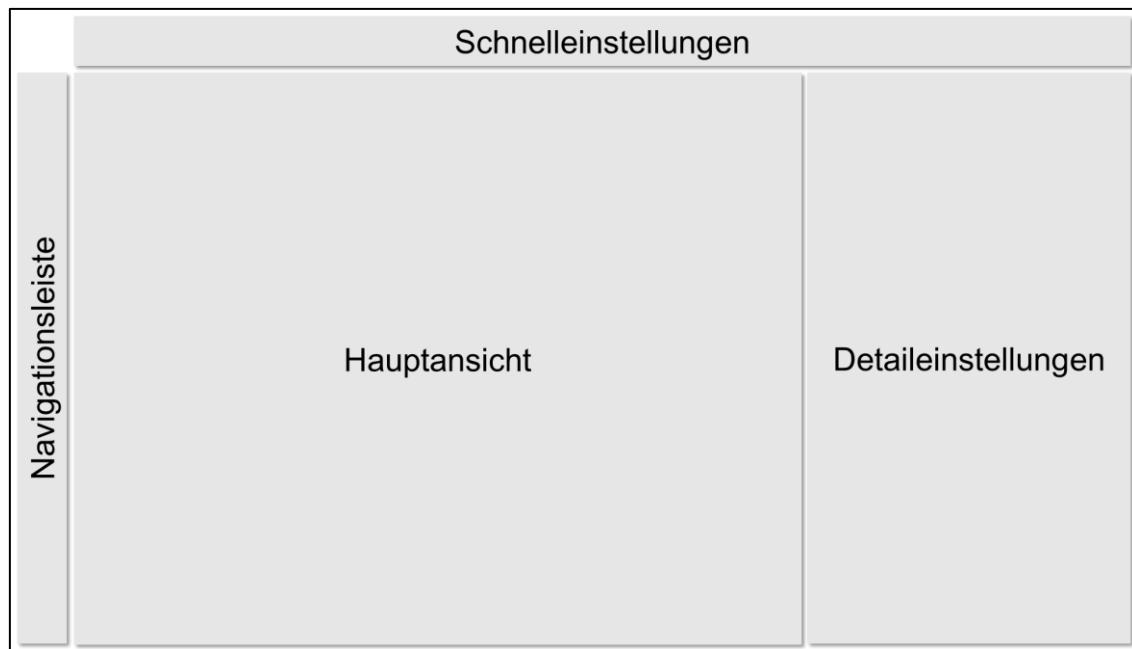


Abbildung 16: Bereiche des Systems

Die Navigationsleiste ermöglicht den Wechsel zwischen fünf verschiedenen Ansichten, wobei die Dashboard-Ansicht wiederum fünf Panels enthält. Tabelle 11 listet diese zehn Elemente auf und verweist auf die Kapitel, in denen die Elemente jeweils näher erläutert werden.

Ansicht	Verweis	Panel	Verweis
Dashboard	Kapitel 7.2.1	3D-Ansicht	Kapitel 7.3.1
Parameter verwalten	Kapitel 7.2.2	2D-Ansicht	Kapitel 7.3.2
Bericht erstellen	Kapitel 7.2.3	Systemstatus	Kapitel 7.3.3
Datenquelle ändern	Kapitel 7.2.4	Kennzahlen	Kapitel 7.3.4
Einstellungen	Kapitel 7.2.5	Parameter-Übersicht	Kapitel 7.3.5

Tabelle 11: Übersicht über Bereiche und Panels des Systems

7.2 Ansichten

In den folgenden Unterkapiteln werden die in Tabelle 11 aufgelisteten Ansichten des Systems anhand von Mockups dargestellt und näher erläutert. Mit der linken Navigationsleiste, die in allen Ansichten sichtbar ist, kann zwischen den Ansichten gewechselt werden. Standardmäßig wird das Dashboard angezeigt.

7.2.1 Dashboard

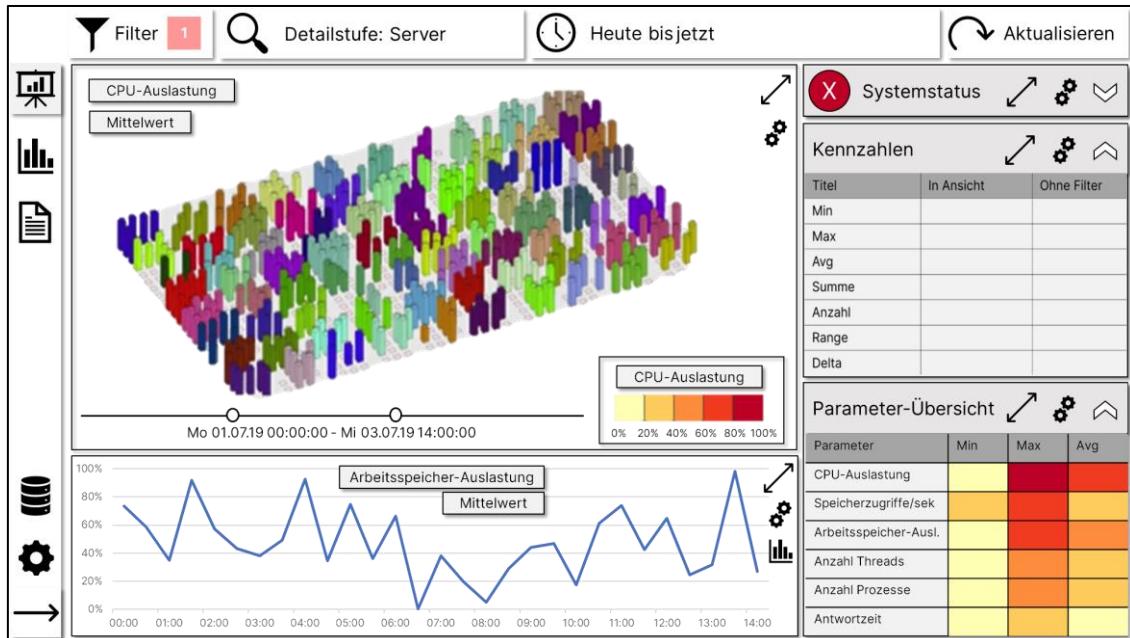


Abbildung 17: Mockup – Dashboard

Das Dashboard (Abbildung 17) bildet das zentrale Element des Systems. Dort werden alle relevanten Informationen angegeben, die fürs Monitoring benötigt werden. Diese Informationen werden in fünf Panels dargestellt (vgl. Kapitel 7.3). Oberhalb der Diagramme wird eine Leiste mit Schnelleinstellungen angezeigt. Diese ermöglichen es Filter zu setzen (vgl. Kapitel 7.4.1), zwischen verschiedenen Detailstufen zu zoomen (vgl. Kapitel 7.4.2), Daten zu einem bestimmten Zeitpunkt oder -intervall anzuzeigen (vgl. Kapitel 7.4.3) und die Diagramme auf Basis getroffener Einstellungen zu aktualisieren. Darüber hinaus kann für jedes Panel eine Sidebar mit Detaileinstellungen geöffnet werden (vgl. Kapitel 7.4.5).

7.2.2 Parameter verwalten

Die Diagramme im Dashboard können verschiedene Parameter visualisieren. Alle zur Verfügung stehenden Parameter können untenstehender Ansicht (Abbildung 18) konfiguriert werden. Über das Stift-Icon können detailliertere Einstellungen für einen einzelnen Parameter geöffnet werden (Abbildung 19).

In der Spalte Sichtbarkeit können einzelne Parameter für Dropdown-Menüs ausgebldet werden. Dies bietet sich an, falls sie derzeit nicht von Interesse sind, aber nicht gänzlich gelöscht werden sollen. Der Titel des Parameters kann nur in den Detaileinstellungen geändert werden. Die Minimum- und Maximum-Werte geben die Grenzwerte an, die für den jeweiligen Parameter vorkommen können. Bei der CPU-Auslastung sind diese beispielsweise auf 0% bis 100% vorgegeben, während sie sich bei ‚Anzahl Anfragen‘ an den höchsten Wert in der Datenbank anpassen. Für die Diagramme im Dashboard kann eine Farbskala aktiviert werden, welche individuell konfiguriert werden kann. Voreingestellt sind eine lineare und eine logarithmische Skala. Der Benutzer hat darüber hinaus die Möglichkeit, eine eigene Skala zu hinterlegen. Die Farbwerte sind standardmäßig fünfstufig angelegt (vgl. Kapitel 7.6.4), können aber beliebig erweitert oder verändert werden. Mit dem Müllheimer-Icon können Parameter gelöscht und über den untenstehenden Button können neue Parameter hinzugefügt werden. Mit dem Button oben rechts können Queries eingeblendet werden, die es ermöglichen, vorhandene und eigene Parameter mit der angebundenen Datenquelle zu verknüpfen (Abbildung 18).

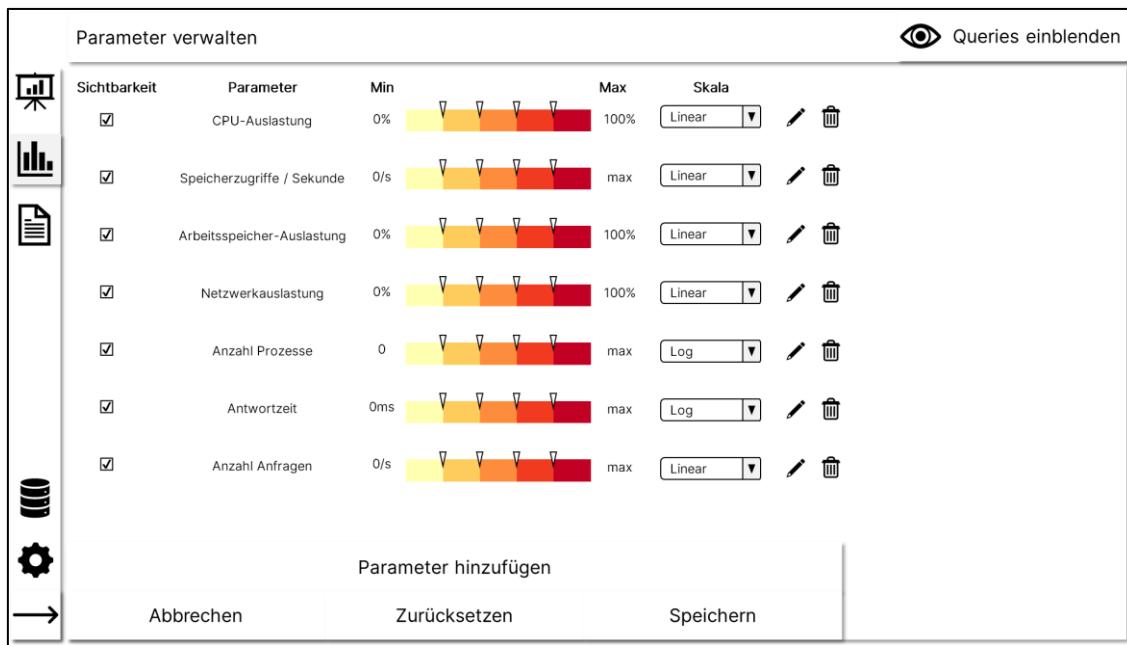


Abbildung 18: Mockup – Parameter verwalten

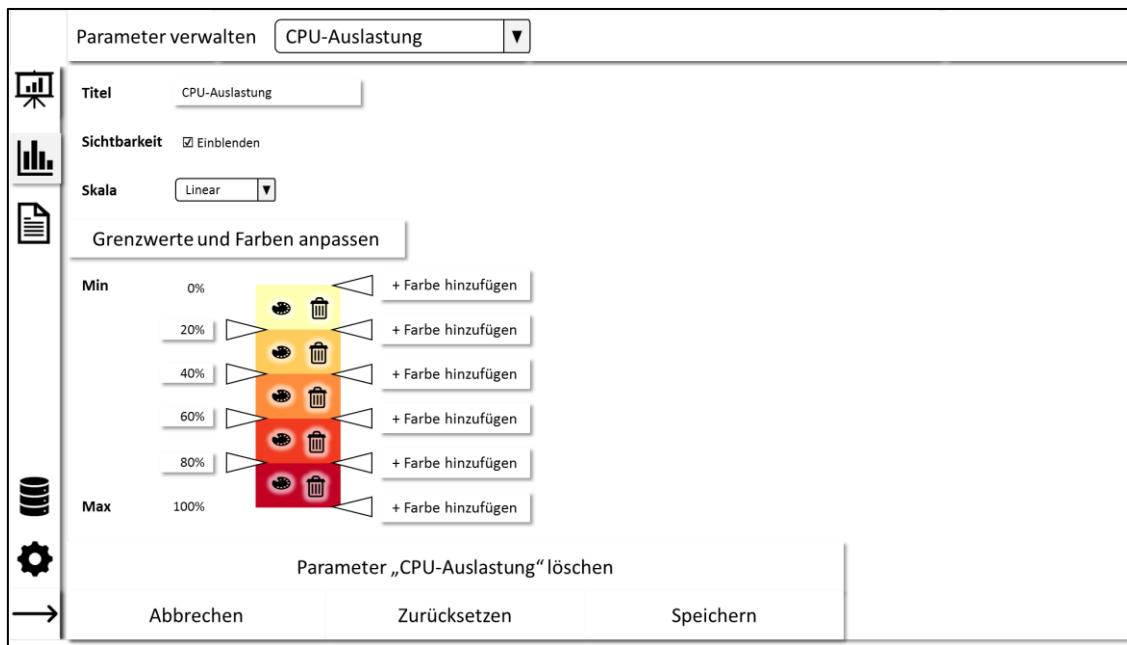


Abbildung 19: Mockup – Parameter verwalten – Detaileinstellungen

In den Detaileinstellungen können weitere Farben ergänzt oder vorhandene geändert bzw. gelöscht werden. Außerdem lassen sich die Grenzwerte der Farben manuell ändern (Abbildung 19).

7.2.3 Bericht erstellen

Um vergangene Ereignisse, kritische Zustände oder berechnete Prognosen im Unternehmen zu kommunizieren oder diese zu speichern, kann ein Bericht erzeugt und als PDF-Dokument exportiert werden (Abbildung 20).

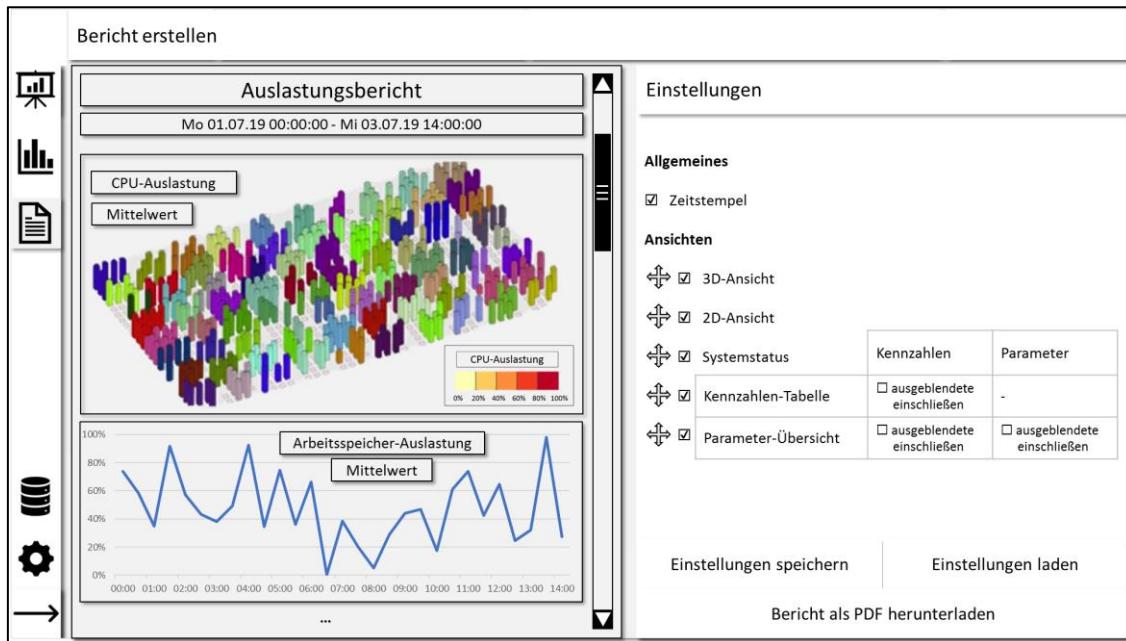


Abbildung 20: Mockup – Bericht erstellen

In den Einstellungen kann ausgewählt werden, welche Elemente in welcher Reihenfolge im Bericht enthalten sein sollen. Links ist eine Vorschau des Berichtes zu sehen. Die Diagramme sind mit dem Dashboard verlinkt und werden im Bericht genauso angezeigt, wie im Dashboard. Zusätzlich ermöglichen die Einstellungen es, Kennzahlen und Parameter anzuzeigen, die im Dashboard ausgeblendet sind. Die getroffenen Einstellungen können gespeichert und geladen werden (Abbildung 20).

7.2.4 Datenquelle ändern

Im System können verschiedene Datenquellen angebunden werden. Beispielsweise werden CSV-Dateien, Solr und Prometheus unterstützt. Dabei können sowohl rein historische als auch live Daten verwendet werden. Die Tabelle rechts in der Abbildung zeigt eine Vorschau der gewählten Datenquelle. Darüber lassen sich auch einzelne Spalten bestimmten Parametern zuordnen (Abbildung 21).

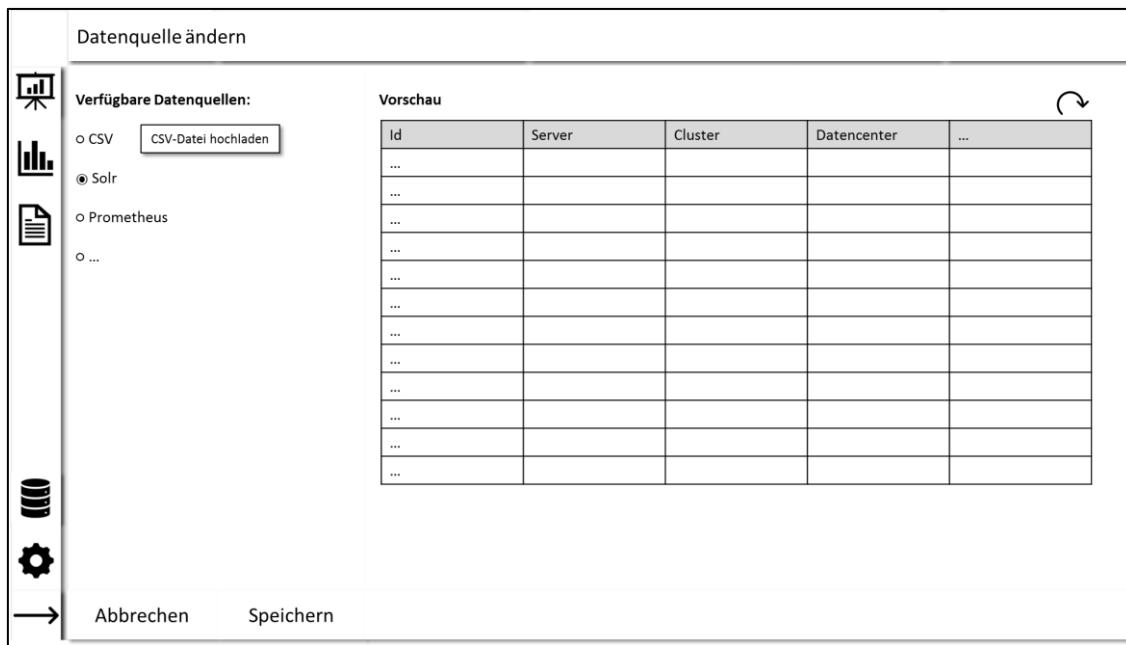


Abbildung 21: Mockup – Datenquelle ändern

7.2.5 Einstellungen

Die in untenstehender Ansicht getroffenen Einstellungen wirken sich global aus, während Einstellungen innerhalb von Diagrammen nur diese betreffen. Die im Mockup aufgelisteten Einstellungen wurden zur Veranschaulichung in grün nummeriert und werden unterhalb des Mockups erläutert (Abbildung 22).

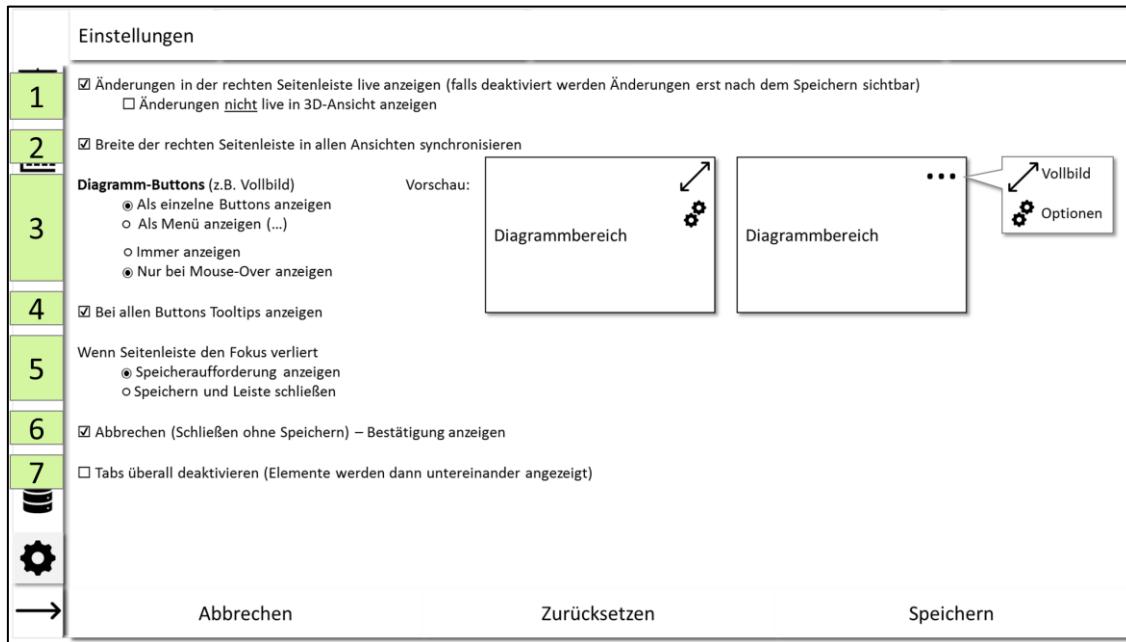


Abbildung 22: Mockup – Einstellungen

1. Standardmäßig werden Änderungen im Dashboard in der Detaileinstellungen-Seitenleiste (vgl. Kapitel 7.4.5) direkt in der Hauptansicht sichtbar. Diese werden dann beim Speichern beibehalten und beim Abbrechen zurückgesetzt. Mit Option 1 in den Einstellungen kann geändert werden, dass Änderungen erst beim Speichern sichtbar werden. Im Punkt darunter können live Änderungen in der 3D-Ansicht deaktiviert werden. Dies bietet sich bei einem System mit schlechter Performance an, um zu vermeiden, dass die 3D-Balken bei jeder Änderung neu geladen werden müssen.
2. Die Breite der Detaileinstellungen-Seitenleiste kann mit Hilfe der Maus geändert werden. Wenn Option 2 ausgewählt ist, wird diese manuelle Änderung der Breite auf die Detaileinstellungen aller Panels angewendet. Wenn die Option deaktiviert ist, können für jedes Panel eigene Breiteneinstellungen getroffen werden.
3. Mit Option 3 können die Buttons oben rechts in den Panels verändert werden. Daneben ist eine Vorschau angezeigt, wie die aktuell getroffenen Einstellungen aussehen. Die Panels haben jeweils mehrere Buttons. Es kann ausgewählt werden, ob diese alle einzeln oder in einem Menü (dargestellt durch drei Punkte) angezeigt werden. Zusätzlich kann festgelegt werden, ob die Buttons immer oder nur bei einem Mouse-Over angezeigt werden sollen.
4. Wenn Option 4 ausgewählt ist, wird bei allen Buttons bei einem Mouse-over jeweils in einem Tooltip angezeigt, welche Funktion der Button erfüllt.
5. Mit Option 5 kann festgelegt werden, was passieren soll, wenn der Benutzer bei den Detaileinstellungen weder ‚Speichern‘ noch ‚Abbrechen‘ anklickt, sondern stattdessen andere Ansichten öffnet oder Aktionen ausführt. Zur Auswahl steht, dass der Benutzer ein Popup mit einer Speicheraufforderung angezeigt bekommt oder dass Änderungen automatisch gespeichert werden.

6. Falls der Benutzer in den Detaileinstellungen Änderungen vorgenommen hat und auf ‚Abbrechen‘ klickt, kann in Option 6 festgelegt werden, ob die Änderungen direkt verworfen werden sollen, oder ob der Benutzer ein Popup angezeigt bekommt, in dem er bestätigen muss, dass er die Änderungen nicht speichern möchte.
7. Das ganze System ist so konzipiert, dass es auf einem Bildschirm angezeigt werden kann, ohne dass gescrollt werden muss. Bei kleinen Bildschirmen, mobilen Geräten oder entsprechend eigener Vorlieben kann mit dieser Option umgestellt werden, dass in den Detaileinstellungen keine Tabs, sondern alle Einstellungen untereinander angezeigt werden.

7.3 Panels

In diesem Kapitel werden die in Tabelle 11 aufgelisteten Panels des Dashboards sowie deren Detaileinstellungen näher erläutert. Zur Veranschaulichung wurden einige Elemente in den Mockups nummeriert. Auf diese wird sich im Text mit in Klammern angegebenen Zahlen bezogen. Alle Panels besitzen oben rechts Buttons, um sie im Vollbild anzuzeigen und die Seitenleiste mit Detaileinstellungen zu öffnen. In den allgemeinen Einstellungen kann festgelegt werden, in welcher Form und wann diese Buttons angezeigt werden sollen (vgl. Kapitel 7.2.5).

7.3.1 3D-Ansicht

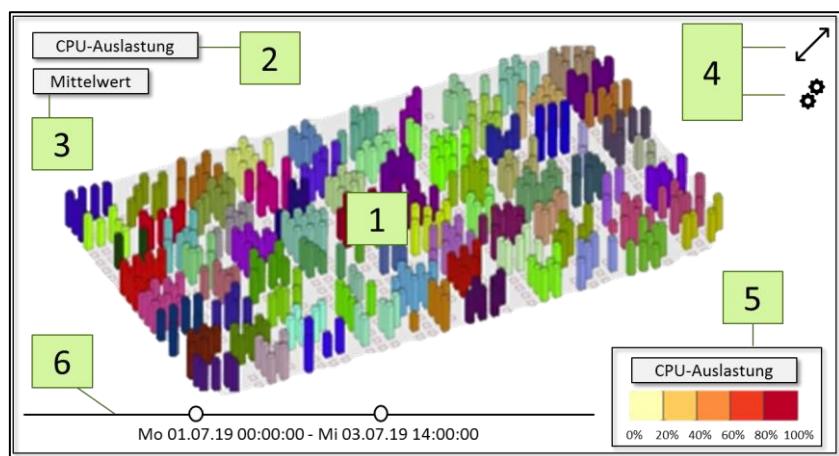


Abbildung 23: Mockup – 3D-Ansicht

Dieses Panel und konkret die dreidimensionale Darstellung (1) stellt das zentrale Element des Systems dar. Es kann umgeschaltet werden, welcher Parameter auf die Höhe der Balken (2) und welcher auf die Farben angewandt werden soll (5). Entsprechend ändert sich die Legende für die Farben (5) und die Skalierung der z-Achse (Höhe der Balken) (1). Es kann zwischen allen Parametern gewechselt werden, die in Bereich ‚Parameter verwalten‘ aktiviert sind (vgl. Kapitel 7.2.2).

Falls ein Zeitbereich ausgewählt wurde, kann eingestellt werden, wie die Werte des Bereichs zusammengefasst werden sollen (3). Zur Auswahl stehen: Mittelwert, Summe, Minimum und Maximum. Der aktuell dargestellte Zeitbereich / -punkt wird unterhalb des Sliders angezeigt (6). Der Slider selbst dient dazu, den oben gewählten Zeitbereich weiter einzuschränken. Falls ein Zeitpunkt ausgewählt wurde, besitzt der Slider nur einen Punkt und die Option zur Auswahl der Art der Aggregation (3) entfällt.

Für den Farbcode (5) kann derselbe Parameter wie für die Balken (2) oder ein anderer gewählt werden. In den Einstellungen existieren weitere Optionen der Färbung.

Mit den Buttons oben rechts (4) kann das Diagramm im Vollbild angezeigt werden und es können Detaileinstellungen festgelegt werden.

Die Detaileinstellungen bestehen auf Grund ihrer Komplexität aus den vier Tabs

- Allgemeine Optionen,
- Abstände & Grenzen,
- Färbung sowie
- Balkenanordnung.

Auf den folgenden Seiten werden diese jeweils im Detail erläutert.

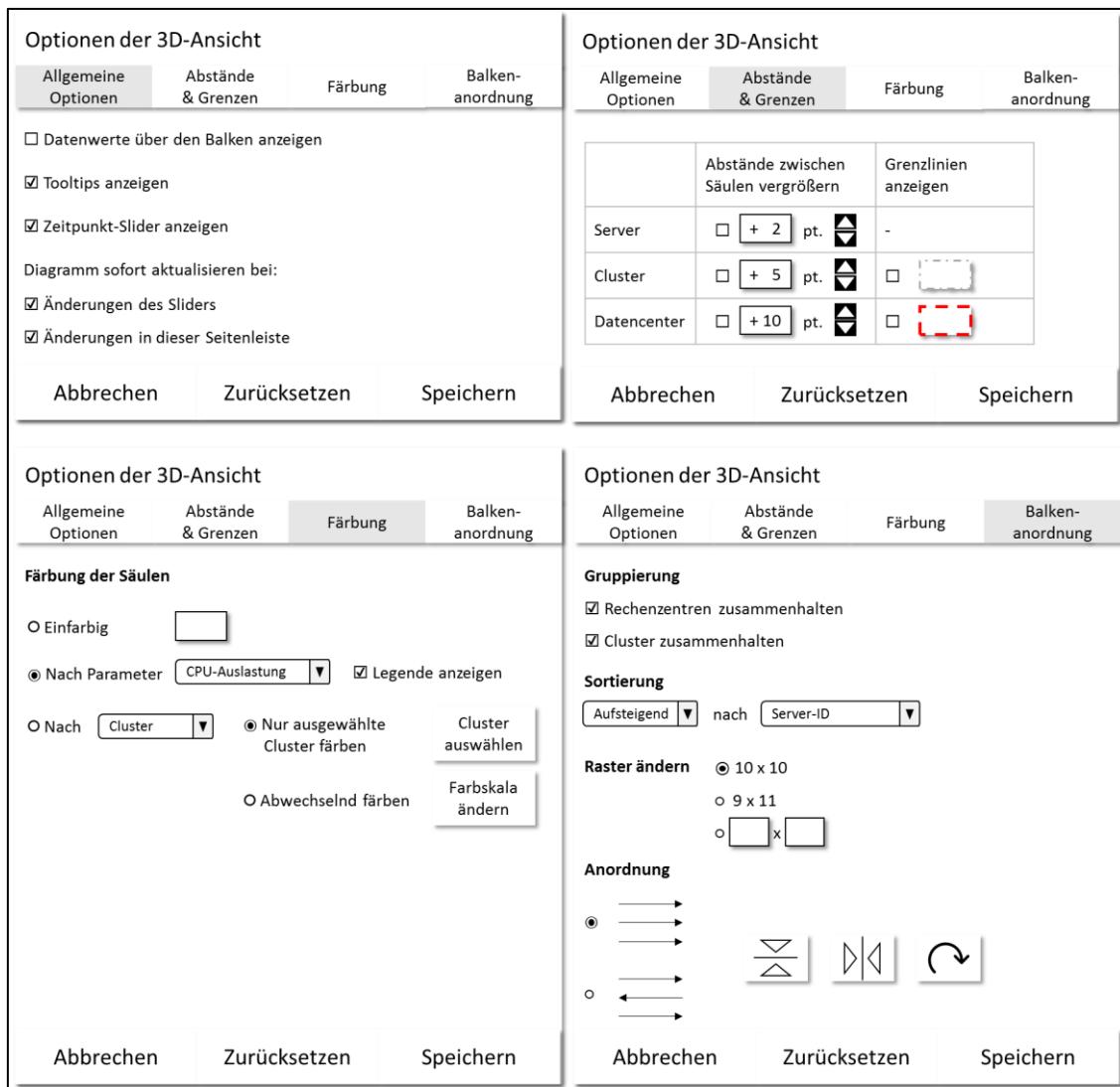


Abbildung 24: Mockup – 3D-Ansicht – Detaileinstellungen

Allgemeine Optionen

In diesem Tab können die Datenwerte für den gewählten Parameter über allen Balken angezeigt werden. Diese Option bietet sich besonders dann an, wenn nur wenige Balken angezeigt werden. Mit der zweiten Option können Tooltips für die Balken aktiviert werden. Diese Tooltips zeigen alle verfügbaren Informationen für den gewählten Balken (Server-, Cluster- und Rechenzentrum-ID sowie die Werte für alle aktivierte Parameter). Mit der dritten Option kann der Slider unterhalb des Diagramms ein- und ausgeblendet werden. Die letzten beiden Optionen ermöglichen es analog zu den allgemeinen Einstellungen umzuschalten, wann das Diagramm aktualisiert werden soll (Abbildung 24 oben links).

Abstände & Grenzen

Mit diesen Optionen können die Abstände zwischen den Balken vergrößert werden. Am Beispiel der Bundesagentur für Arbeit sind die Optionen ‚Server‘, ‚Cluster‘ und ‚Datencenter‘ voreingestellt. Dies ermöglicht es beispielsweise, die Server-Balken eng nebeneinander darzustellen, während zwischen den Clustern ein größerer und zwischen den Datencentern ein noch größerer Abstand ist. Standardmäßig ist zwischen allen Servern ein kleiner Abstand. Möchte man diesen verkleinern, kann eine negative Zahl eingetragen werden. Deaktiviert man die Kästchen bzw. trägt den Wert Null ein, entspricht der Abstand genau den Standardeinstellungen. Zusätzlich können Cluster und/oder Datencenter mit dreidimensional gestrichelten Quadern umgeben werden. Die Farbe der Linien lässt sich individuell festlegen (Abbildung 24 oben rechts).

Färbung

Die 3D-Balken können in drei verschiedenen Weisen eingefärbt werden. Bei der ersten Option werden alle Balken einheitlich in der gewählten Farbe eingefärbt. Die zweite Option ermöglicht es, die Balken anhand eines Parameters entsprechend der festgelegten Skala (vgl. Kapitel 7.2.2) einzufärben. Mit der letzten Option kann farblich hervorgehoben werden, welche Balken zu einem Cluster/Datencenter gehören. Dabei können einzelne Server/Cluster/Datencenter ausgewählt werden oder sie werden abwechseln gefärbt. Letztere Option ist in dem Mockup in Abbildung 17 dargestellt (Abbildung 24 unten links).

Balkenanordnung

In den vorliegenden Mockups sind die X- und Y-Achsen des Diagramms mit keinem Parameter verknüpft. Daher können die Balken nach Belieben angeordnet werden. Dazu kann im letzten Tab die Anordnung, Gruppierung und Sortierung flexibel angepasst werden. Bei letzterer kann zwischen auf- und absteigend in Kombination mit einem beliebigen Parameter gewählt werden. Das Raster gibt an, wie viele Balken neben- und hintereinander angezeigt werden. Voreingestellt sind auf Basis der Anzahl an Balken berechnete Raster, die eine optimale Verteilung auf den angezeigten Bereich bieten. Zusätzlich kann eine eigene Verteilung eingetragen werden (Abbildung 24 unten rechts).

7.3.2 2D-Ansicht

Das zweidimensionale Diagramm zeigt denselben Zeitbereich an, wie das dreidimensionale Diagramm. Falls ein einzelner Zeitpunkt gewählt wurde, bildet dieser die rechte Grenze des Diagramms. In diesem Diagramm können ebenfalls individuell ein Parameter sowie eine Art der Aggregation festgelegt werden. Die Buttons oben rechts vergrößern das Diagramm auf Vollbild, öffnen die Detaileinstellungen und ermöglichen es, den Diagrammtyp zu ändern (Abbildung 25). Letzteres ist ebenfalls über einen Button in den Detaileinstellungen möglich (Abbildung 26 links).

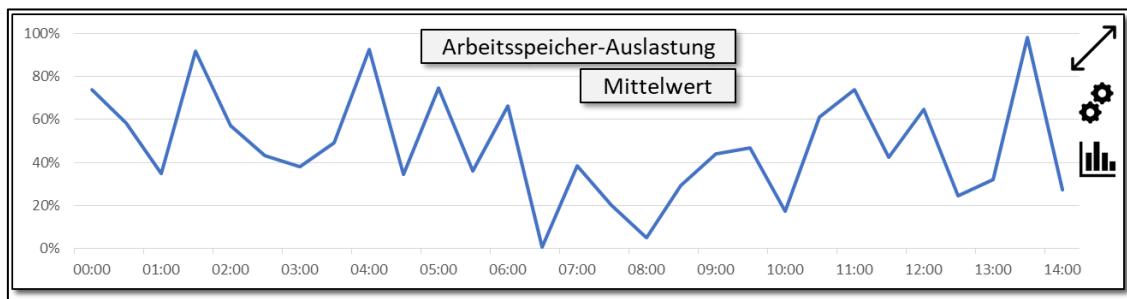


Abbildung 25: Mockup – 2D-Ansicht

Optionen der 2D-Ansicht		Diagrammtyp ändern		
<input type="checkbox"/> Datenwerte anzeigen				
<input checked="" type="checkbox"/> Tooltips anzeigen				
Liniенfarbe:				
		Diagrammtyp ändern		
		<input type="button" value="Abbrechen"/>	<input type="button" value="Zurücksetzen"/>	<input type="button" value="Speichern"/>

Abbildung 26: Mockup – 2D-Ansicht – Detaileinstellungen

In den Detaileinstellungen kann eine Farbe für die Linien festgelegt werden. Außerdem können analog zur 3D-Ansicht Datenwerte und Tooltipps aktiviert werden. In einer zusätzlichen Seitenleiste kann der Diagrammtyp zwischen Linien-, Punkt-, Flächen- und Säulendiagramm gewechselt werden. Für die Prognosen bietet sich jedoch der Typ Liniendiagramm am meisten an (Abbildung 26).

7.3.3 Systemstatus

In diesem Panel wird mittels Machine Learning (vgl. Kapitel 3.1.5) angezeigt, welche Server, Cluster oder Rechenzentren in nächster Zeit ausfallen, ungeplant im Leerlauf sind oder ungeplant unter Volllast arbeiten. Für diese Vorfälle wird aufgeführt, mit welcher Wahrscheinlichkeit sie wann und wo eintreten. Links in der Abbildung ist die Ansicht im Dashboard, rechts ein Ausschnitt der Vollbildansicht dargestellt. Das rote Icon ändert sich entsprechend der höchsten Wahrscheinlichkeit. Alle Spalten können mit den kleinen Pfeilen beliebig sortiert und gefiltert werden. In der kompakten Ansicht kann für den Zeitpunkt gewechselt werden, ob er absolut (Datum + Uhrzeit) oder relativ (in x Stunden) angezeigt wird. Das gesamte Panel kann über den weißen Pfeil ein- und ausgeklappt werden (Abbildung 27).

The mockup displays two versions of the 'Systemstatus' panel side-by-side. Both versions feature a red circular icon with a white 'X' at the top left, followed by the title 'Systemstatus'. On the right are three icons: a gear, a double-headed arrow, and a document.

Left View (Compact Dashboard):

%	Zeitpunkt	Art des Vorfalls	Ort des Vorfalls
80%	Datum + Zeit	Leerlauf	Server 17, Cluster 9, RZ 2
75%	In x Stunden	Ausfall	
		Volllast	

Right View (Full View):

%	Zeitpunkt	Art des Vorfalls	Ort des Vorfalls
80%	Datum + Zeit	Leerlauf	Server 17
75%	In x Stunden	Ausfall	Cluster 9
		Volllast	RZ 2

Abbildung 27: Mockup – Systemstatus

Allgemeine Optionen

In den Detaileinstellungen können die Sortier- und Filtersymbole ausgeblendet und deren Einstellungen gespeichert werden. In diesen Optionen kann ebenfalls zwischen absoluten und relativen Zeitpunktangaben gewechselt werden. In der letzten Option kann aktiviert werden, dass auch vergangene Vorfälle (bis zu einem gewissen Zeitpunkt oder alle) angezeigt werden (Abbildung 28 oben).

Spalten

In diesem Bereich können einzelne Spalten jeweils in der kompakten und der Vollbild-Ansicht ausgeblendet sowie per Drag-n-Drop in der Reihenfolge geändert werden (Abbildung 28 Mitte).

Darstellungsoptionen

Hier können die Formate der Zeitpunkte geändert werden. Wird das Kästchen neben der Überschrift deaktiviert, können für beide Ansichten unterschiedliche Formate festgelegt werden. Andernfalls zeigen sie dieselben Formate an. Mit der letzten Option kann festgelegt werden, ob der Ort des Vorfalls immer in Langform (Server + Cluster + Datencenter) oder nur auf Basis der gewählten Detailstufe (z. B. Server) angezeigt werden soll (Abbildung 28 unten).

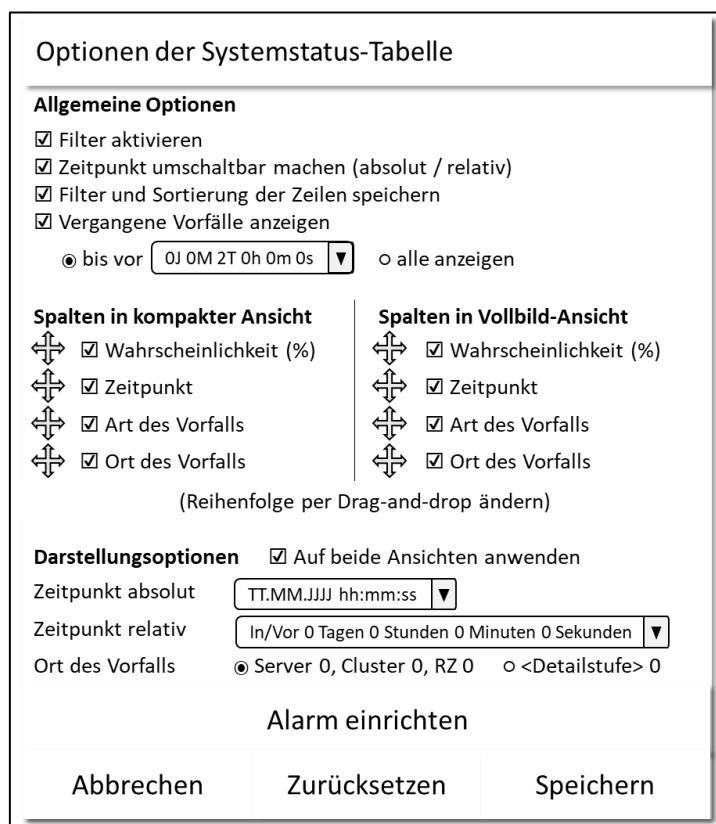


Abbildung 28: Mockup – Systemstatus – Detaileinstellungen

Der untenstehende Button ‚Alarm einrichten‘ ermöglicht es, eigene Regeln festzulegen, wann der Benutzer eine Pushnachricht oder E-Mail erhalten soll. Eine solche Regel könnte zum Beispiel lauten: Sende eine E-Mail an den Administrator, wenn in den nächsten fünf Stunden Cluster 15 eine CPU-Auslastung von über 90% erreicht.

7.3.4 Kennzahlen

Auf Basis des Parameters, der in der 3D-Ansicht angezeigt wird, werden in der nebenstehenden Tabelle die in untenstehender Abbildung aufgelisteten Kennzahlen angezeigt. Diese erweitern die Informationen aus den Diagrammen. Es werden einerseits Kennzahlen auf Basis aktiver Filter und andererseits Kennzahlen ohne Berücksichtigung der Filter angezeigt. Dadurch wird gewährleistet, dass auch bei aktivem Filter extreme Werte außerhalb des Filters erkannt werden können. Mit den Buttons kann die Tabelle im Vollbild angezeigt oder eingeklappt werden und es können die Detaileinstellungen geöffnet werden (Abbildung 29).

Kennzahlen		
Titel	In Ansicht	Ohne Filter
Min		
Max		
Avg		
Summe		
Anzahl		
Range		
Delta		

Abbildung 29: Mockup – Kennzahlen

In den Detaileinstellungen kann festgelegt werden, wie sich die Panels beim Einklappen verhalten sollen. Sie können entweder auf eine maximale Größe von einem Drittel der verfügbaren Höhe limitiert werden, sodass sie alle gleich groß werden, oder jeweils so groß angezeigt werden, wie der verfügbare Platz es ermöglicht. In den Darstellungsoptionen kann eingestellt werden, ob jeweils nur die Kennzahlen für die aktuelle Ansicht oder auch diejenigen ohne Berücksichtigung des Filters angezeigt werden sollen (falls kein Filter eingestellt wurde, blendet sich die ‚ohne Filter‘-Spalte automatisch aus). Im dritten Abschnitt der Einstellungen können Kennzahlen für die kompakte Ansicht ein- und ausgeblendet sowie deren Reihenfolge geändert werden. Neben der Auflistung verfügbarer Kennzahlen steht jeweils eine kurze Erklärung zu ihnen. Abschließend kann ausgewählt werden, ob in der Vollbildansicht auch die deaktivierten Kennzahlen angezeigt werden sollen (Abbildung 30).

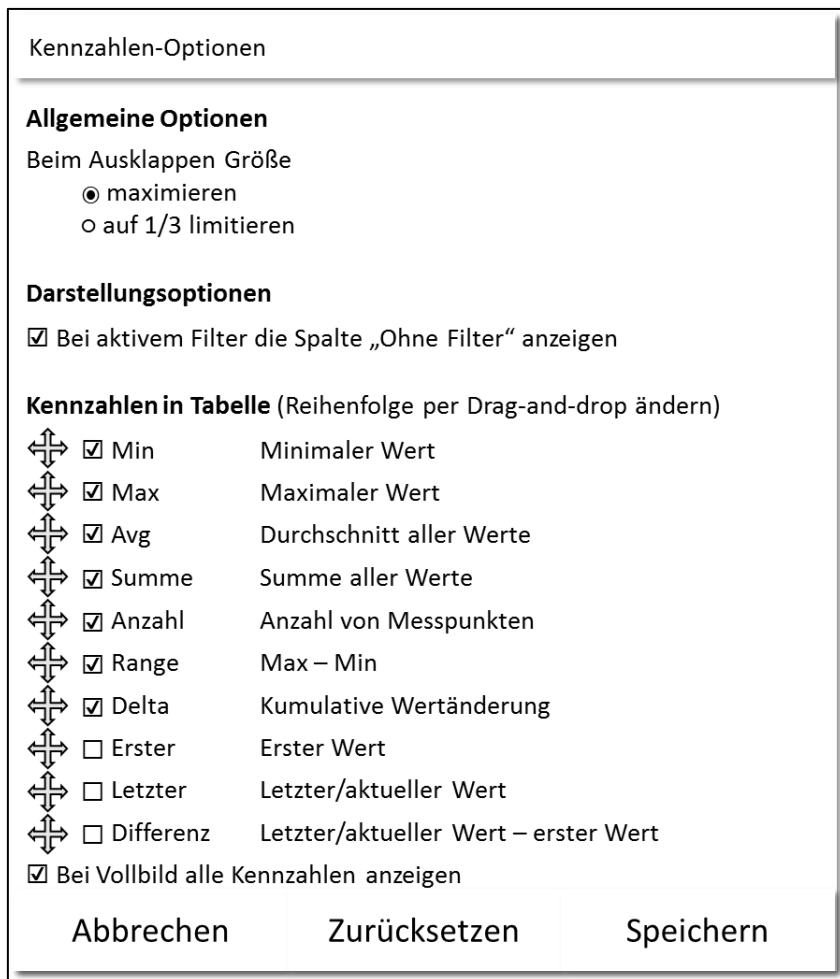


Abbildung 30: Mockup – Kennzahlen – Detaileinstellungen

7.3.5 Parameter-Übersicht

Die Parameter-Übersicht Tabelle zeigt in den Spalten Kennzahlen und in den Zeilen Parameter an. Dies ermöglicht es, kritische Werte für derzeit nicht betrachtete Parameter zu erkennen. Die Tabelle passt sich an die aktivierte Detailstufe an. Der Farbcde unterstützt den Benutzer dabei, schnell kritische Werte zu erkennen (Abbildung 31). Dieser Farbcde sowie die Werte in Textform können in den Detaileinstellungen (de)aktiviert werden. Darüber hinaus kann auch hier das Verhalten beim Einklappen festgelegt werden. Darunter kann festgelegt werden, welche Kennzahlen und Parameter in welcher Reihenfolge angezeigt werden sollen und ob im Vollbild jeweils alle sichtbar sein sollen (Abbildung 32).

Parameter	Min	Max	Avg
CPU-Auslastung			
Speicherzugriffe/sek			
Arbeitsspeicher-Ausl.			
Anzahl Threads			
Anzahl Prozesse			
Antwortzeit			

Abbildung 31: Mockup – Parameter-Übersicht

Parameter-Übersicht-Optionen

Allgemeine Optionen

Beim Ausklappen Größe

- maximieren
- auf 1/3 limitieren

Werte anzeigen

Farbcodierung verwenden

Kennzahlen (Spalten)	Parameter (Zeilen)
<input checked="" type="checkbox"/> Min (minimaler Wert)	<input checked="" type="checkbox"/> CPU-Auslastung
<input checked="" type="checkbox"/> Max (maximaler Wert)	<input checked="" type="checkbox"/> Speicherzugriffe/Sekunde
<input checked="" type="checkbox"/> Avg (durchschnittl. Wert)	<input checked="" type="checkbox"/> Arbeitsspeicher-Ausl.
<input checked="" type="checkbox"/> Median	<input checked="" type="checkbox"/> Anzahl Threads
<input checked="" type="checkbox"/> Bei Vollbild alle Kennzahlen anzeigen	<input checked="" type="checkbox"/> Bei Vollbild alle Parameter anzeigen

(Reihenfolge per Drag-and-drop ändern)

Ausgeblendete Parameter können in der Parameter-Verwaltung eingeblendet werden

Abbrechen Zurücksetzen Speichern

Abbildung 32: Mockup – Parameter-Übersicht – Detaileinstellungen

Angezeigt werden in der Übersicht nur Parameter, die im Bereich ‚Parameter verwalten‘ (Kapitel 7.2.2) aktiviert sind. Um Änderungen schnell vornehmen zu können, ist der entsprechende Bereich in den Detaileinstellungen zusätzlich verlinkt (Abbildung 32).

7.4 Weitere Elemente

In den folgenden drei Unterkapiteln werden die Elemente ‚Filter‘, ‚Detailstufenzoom‘ und ‚Datumsauswahl‘ der Schnelleinstellungen näher erläutert. Mit dem Button oben rechts können alle Panels manuell aktualisiert werden (Abbildung 33). Darauf folgen ein Kapitel zur Aktivierung und Darstellung von Prognosen sowie zur allgemeinen Funktionsweise der Detaileinstellungen-Seitenleiste.



Abbildung 33: Mockup – Schnelleinstellungen

7.4.1 Filter

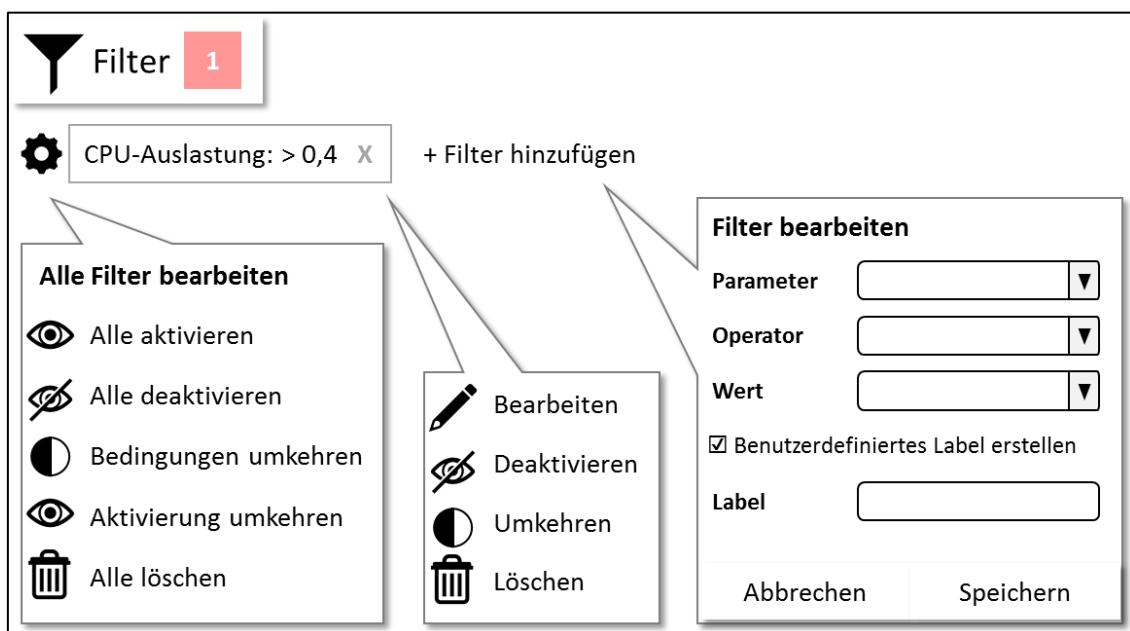


Abbildung 34: Mockup – Filter

Um die Anzahl an Balken zu reduzieren, können beliebig viele Filter gesetzt werden. Neben dem entsprechenden Button zeigt eine rot hinterlegte Zahl an, wie viele Filter derzeit aktiv sind. Beim Klick auf den Button erscheint unter den Schnelleinstellungen eine Leiste mit allen Filtern. Falls derzeit keine Filter aktiv sind, ist die Zahl nicht sichtbar und ein Klick auf den Filter-Button öffnet direkt das Menü, um einen neuen Filter zu erstellen (Abbildung 34).

Über das Zahnrad-Icon können Einstellungen für alle Filter gleichzeitig getätigt werden. Daneben werden alle Filter jeweils in einem eigenen Kasten angezeigt. Über das kleine „x“ rechts können diese gelöscht oder durch einen Klick auf den Kasten bearbeitet werden. Über den rechten Text „+ Filter hinzufügen“ können neue Filter hinzugefügt werden. Diese lassen sich mit einem individuellen Label versehen. So könnte man sich in der zweiten Zeile zum Beispiel statt „CPU-Auslastung: > 0,4“ anzeigen lassen „CPU > 40%“. Darüber hinaus ist es auch möglich, nur ausgewählte Server/Cluster/Rechenzentren anzuzeigen (Abbildung 34).

7.4.2 Detailstufenzoom

In den vorliegenden Mockups stehen die 3D-Balken jeweils für einzelne Server. Um größere Zusammenhänge zu erkennen, kann es hilfreich sein, die Komplexität zu reduzieren. Dazu kann die Detailstufe verringert werden. Wird beispielsweise „Cluster“ ausgewählt, werden alle Server eines Clusters auf Basis der aktiven Art der Aggregation zu jeweils einem Cluster-Balken zusammengefasst. Zusätzlich passen sich auch die Tabellen für Kennzahlen und Parameter entsprechend der gewählten Detailstufe an (Abbildung 35).

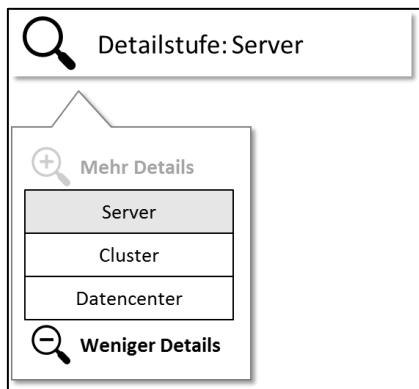


Abbildung 35: Mockup – Detailstufenzoom

7.4.3 Datumsauswahl

Um einen konkreten Zeitraum oder Zeitpunkt auszuwählen, existieren jeweils mehrere Auswahlmöglichkeiten. Mit einem Klick auf das Uhr-Icon öffnet sich die Schnellauswahl (Abbildung 36), ein Klick auf „Heute“ oder „jetzt“ öffnet die komplexe Auswahl für Start- und Endzeitpunkt (Abbildung 37).

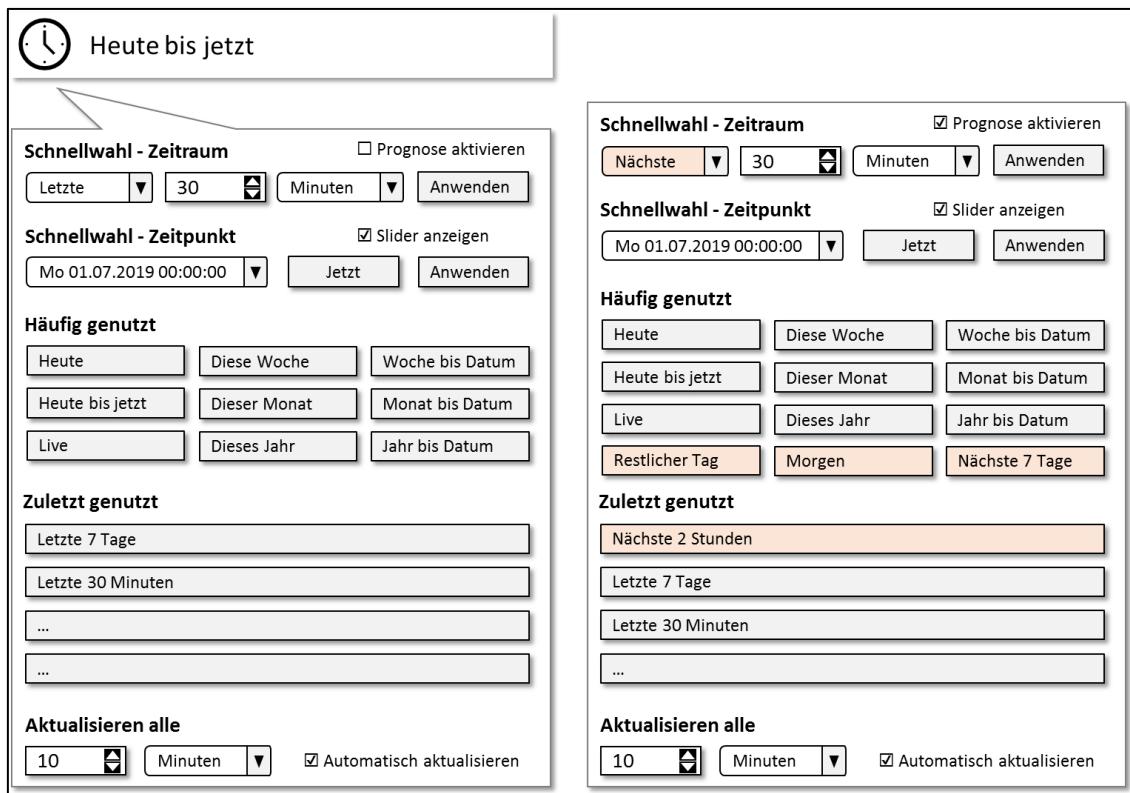


Abbildung 36: Mockup – Datumsauswahl – Schnellauswahl

Schnellauswahl

In der Schnellauswahl kann ein relativer Zeitraum oder ein Zeitpunkt eingetragen werden. Die einzelnen Buttons darunter ermöglichen es, die am häufigsten genutzten Zeiträume und Zeitpunkte sowie die zuletzt verwendeten mit einem Klick zu setzen. Ganz unten kann festgelegt werden, ob und wie oft sich das Dashboard automatisch aktualisieren soll. Dies hat bei absoluten Zeiträumen in der Vergangenheit keinen Effekt, ist jedoch relevant für relative Einstellungen sowie wenn der aktuelle Zeitpunkt Bestandteil der Auswahl ist. Im oberen Teil des Menüs kann der Slider unterhalb der 3D-Ansicht ein- und ausgeblendet werden. Diese Einstellung ist mit der Einstellung aus den Detaileinstellungen der 3D-Ansicht verknüpft (Abbildung 36 links).

Oben rechts kann eine Checkbox für die Prognose aktiviert werden (vgl. Kapitel 7.4.4). Dadurch können auch Zeiträume und Zeitpunkte in der Zukunft angezeigt werden (Abbildung 36 rechts).

Komplexe Auswahl

Die komplexe Auswahl bietet vor allem für die Einstellung von Zeiträumen große Flexibilität. Es kann jeweils für Beginn und Ende des gewünschten Zeitraumes ein Zeitpunkt absolut (2), relativ (3) oder live (4) eingestellt werden. Diese Optionen stehen in drei Tabs zur Verfügung und sind in untenstehender Abbildung dargestellt. Geöffnet wird das Fenster mit den drei Tabs durch Klick auf Beginn oder Ende des Zeitraums (1). Im vorliegenden Mockup wären das ‚vor 30 min.’ und ‚Jetzt‘. Bei der absoluten Zeitangabe (2) kann ein genauer Zeitpunkt mittels Datum und Uhrzeit fest gewählt werden. Bei der relativen Zeitangabe (3) kann ein Zeitpunkt ausgesucht werden, der sich eine bestimmte Zeit vor oder bei aktiver Prognose auch nach der aktuellen Zeit befindet. Darüber hinaus kann als Beginn oder Ende die aktuelle Zeit verwendet werden (4). Diese Zeitangaben (3 und 4) aktualisieren sich nach der eingestellten Zeit automatisch. Alternativ kann die automatische Aktualisierung deaktiviert werden. Dann muss der Benutzer oben rechts in den Schnelleinstellungen manuell auf ‚Aktualisieren‘ klicken (Abbildung 37).

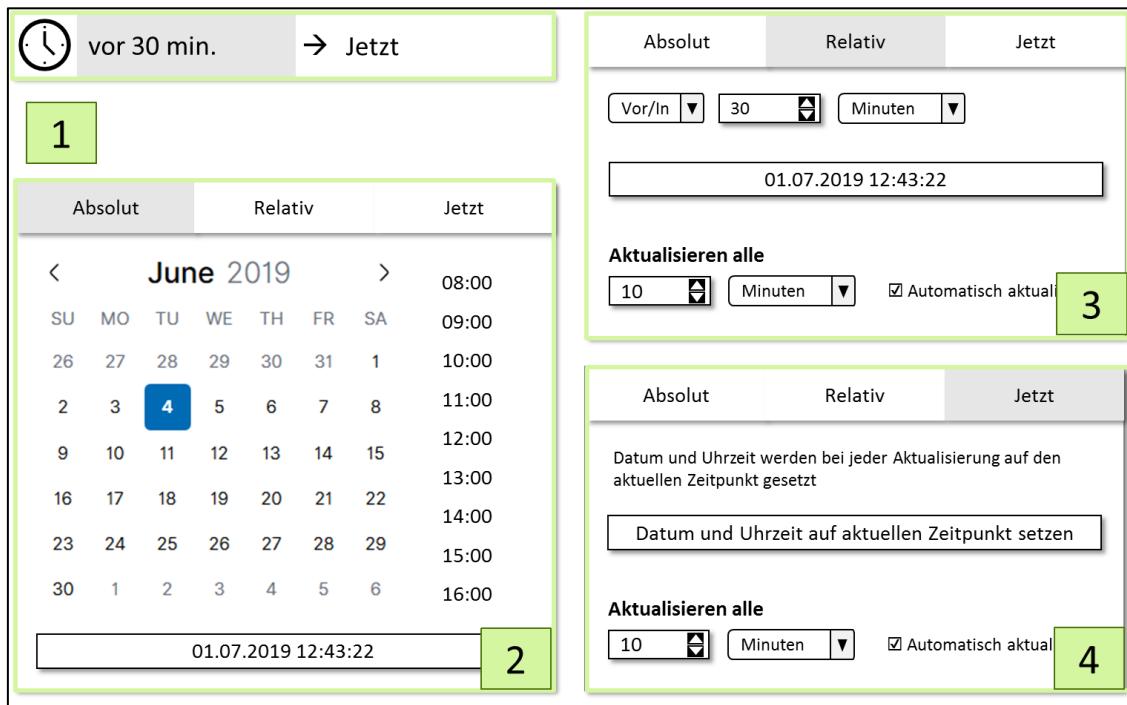


Abbildung 37: Mockup – Datumsauswahl – komplexe Auswahl

7.4.4 Prognose

Falls in der Datumsauswahl die Prognose aktiviert wurde, können Zeiträume und Zeitpunkte in der Zukunft ausgewählt werden. Berechnet werden diese Werte mittels Machine Learning.

Dargestellt werden prognostizierte Werte wie in untenstehender Abbildung am Beispiel der 2D-Ansicht. Dabei erweitert sich der farbige Bereich entsprechend der Unsicherheit der Prognosewerte (Abbildung 38).

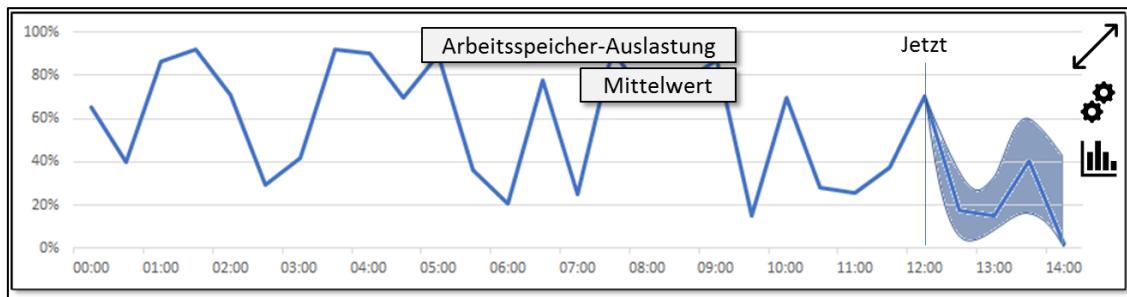


Abbildung 38: Mockup – 2D-Ansicht – Prognose

7.4.5 Detaileinstellungen

Diese Seitenleiste kann für alle Panels des Dashboards geöffnet werden und ermöglicht individuelle Einstellungen. Die folgende Tabelle listet die Seitenleisten auf und verweist auf die jeweiligen Mockups (Tabelle 12). Die Breite der Seitenleiste kann mit der Maus geändert werden. In den allgemeinen Einstellungen lässt sich diese Breitenänderung auf alle Detaileinstellungen-Seitenleisten gleichzeitig anwenden.

Seitenleiste	Verweis
3D-Ansicht	Abbildung 24
2D-Ansicht	Abbildung 26
Systemstatus	Abbildung 28
Kennzahlen	Abbildung 30
Parameter-Übersicht	Abbildung 32

Tabelle 12: Übersicht über Detaileinstellungen-Seitenleisten

7.5 Funktionalität

In den folgenden Kapiteln wird über das Design hinausgehend beschrieben, welche besonderen Funktionen das System erfüllen soll.

7.5.1 Bedienbarkeit mit Tastatur und Maus

Sämtliche Funktionen sollen sowohl per Maus als auch per Tastatur genutzt werden können. Dadurch können Benutzer für jede Funktion die für sie effizienteste oder angenehmste Bedienmethode wählen. Durch Buttons und Slider bei Zahleneingaben kann die Maus für alle Funktionen verwendet werden. Lediglich bei Freitextfeldern wie eigene Labels muss eine virtuelle Tastatur eingeblendet werden. Bei der Bedienung mittels Tastatur werden Tastenkombinationen verwendet. Diese sollen von dem verwendeten System oder bei einer Eigenentwicklung von den infrage kommenden Systemen übernommen werden (vgl. z. B. Kibana [ElKe2019], Grafana [GrKe2019]).

7.5.2 Panels verschieben

In den Einstellungen können alle Panels fixiert oder gelöst werden. Beim Fixieren behalten sie ihre Position, können aber weiterhin im Vollbild geöffnet werden. Ein versehentliches Verschieben wird durch diesen Zustand verhindert. Wenn die Panels gelöst sind, können sie per Drag and Drop neu angeordnet werden. Es wird empfohlen, die Ansicht abschließend wieder zu fixieren, um ein versehentliches Verschieben zu verhindern. Die manuelle Neuanordnung der Panels ermöglicht es den Benutzern, die Ansicht an eigene Bedürfnisse anzupassen und häufiger verwendete Panels prominenter zu platzieren. Die Standardansicht ist jedoch bereits so ausgelegt, dass sie in den meisten Fällen beibehalten werden kann.

7.5.3 Zoomen und Schwenken

Die 3D-Ansicht kann beliebig gezoomt und geschwenkt werden. Beide Optionen sind sowohl mit Maus als auch mit Tastatur möglich. Das jeweilige Vorgehen wird in untenstehender Tabelle beschrieben (Tabelle 13).

	Maus	Tastatur
Zoomen	Mit dem Scrollrad kann hinein- und hinausgezoomt werden.	Tastenkombination: ,Strg' + ,+' (hineinzoomen) ,Strg' + ,-' (hinauszoomen)
Schwenken	Eine beliebige Stelle in der dreidimensionalen Ansicht außerhalb der Balken anklicken und den Mauszeiger bei gedrückter linker Maustaste in die gewünschte Richtung bewegen.	Mit Hilfe der Pfeiltasten kann die Ansicht bei aktivem Fokus beliebig in die gewünschte Richtung geschwenkt werden.

Tabelle 13: Zoomen und Schwenken

7.5.4 Fokus

Es steht immer ein Panel und ein Element innerhalb des Panels im Fokus. Innerhalb des Panels erhält das zuletzt verwendete Element den Fokus. Wurde in der aktuellen Sitzung noch kein Element verwendet, liegt der Fokus auf dem ersten Element innerhalb des Panels. Beim Einsatz der Maus ist dasjenige Panel im Fokus, welches zuletzt angeklickt wurde. Beim Einsatz der Tastatur kann der Fokus mit ,Strg' + ,Tab' zwischen den Panels und mit ,Tab' innerhalb der Elemente des aktiven Panels gewechselt werden.

7.6 Styleguide

In diesem Kapitel sollen grundsätzliche Designvorgaben und Gestaltungshinweise gegeben werden. Dazu folgt zunächst ein Unterkapitel zum allgemeinen Aufbau der Oberfläche, in dem Abstände und Größen erläutert werden. Anschließend wird auf Design und Verwendung von Buttons und Icons sowie auf Schriftgrößen und -arten eingegangen. Abschließend werden die verwendeten Farben erläutert.

Wie auch bei verbreiteten Tools wie Kibana und Grafana werden in der Hauptansicht des Systems Panels eingesetzt. Diese bieten durch die leichte Anpassbarkeit eine gute Nutzerfreundlichkeit und Wartbarkeit. Darüber hinaus sind es die Benutzer gewohnt, eine solche Oberfläche zu bedienen, wodurch die Nutzung des neuen Systems erleichtert wird [Graf2019, Kiba2019].

7.6.1 Allgemeiner Aufbau

Im Folgenden werden die Größenverhältnisse zwischen den Elementen in den Mockups erläutert. Die Navigationsleiste nimmt 5% der Breite, die Schnelleinstellungen nehmen 5% der Höhe ein. Als zentrales Element erhält die Hauptansicht 65% der Breite und die nebenstehenden Elemente bzw. die Detaileinstellungen erhalten die restlichen 30%. In der Höhe erstreckt sich die Hauptansicht standardmäßig über 90% der verfügbaren Höhe. Bei eingeblendetem Filterleiste erhält diese 5% der Höhe, während sich die restlichen Elemente anteilig um die Höhe der Filterleiste verkleinern (Abbildung 39 und Abbildung 40).

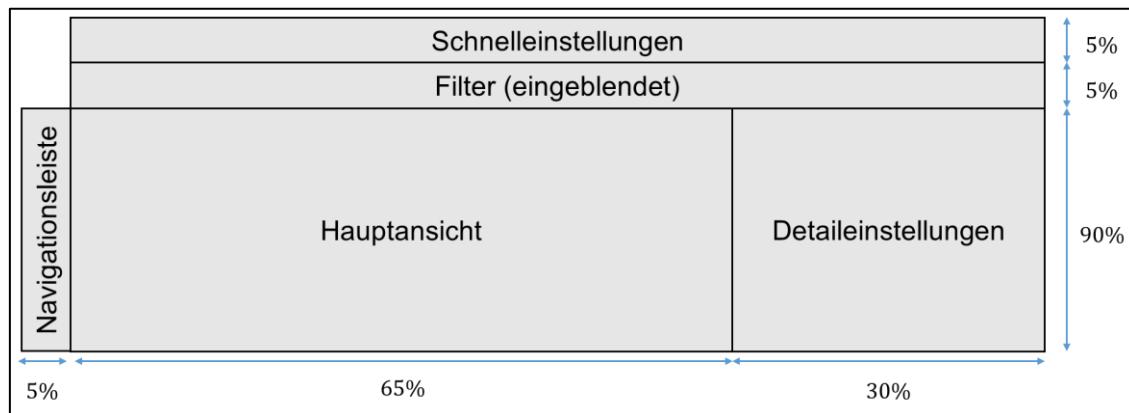


Abbildung 39: Dashboard – Größenübersicht grob

Die einzelnen Elemente der Schnelleinstellungen sollen so skaliert sein, dass die Breite von Filter, Detailstufenzoom und Aktualisieren-Button jeweils maximiert wird. Die Datumsauswahl würde sich in dem Fall auf den restlichen freien Platz strecken. In den Mockups sind die Breiten so aufgeteilt, wie in der detaillierten Größenübersicht dargestellt. Die Hauptansicht bzw. die Panel erstrecken sich über 90% der Höhe und Breite, wobei die 3D-Ansicht als zentrales Element 65% der Höhe und Breite einnimmt. Um den Zeitverlauf gut darzustellen, ist die 2D-Ansicht bei geringerer Höhe ebenso breit. Die rechten Elemente sind eingeklappt 5% hoch und teilen sich ausgeklappt den restlichen verfügbaren Platz (Abbildung 40).

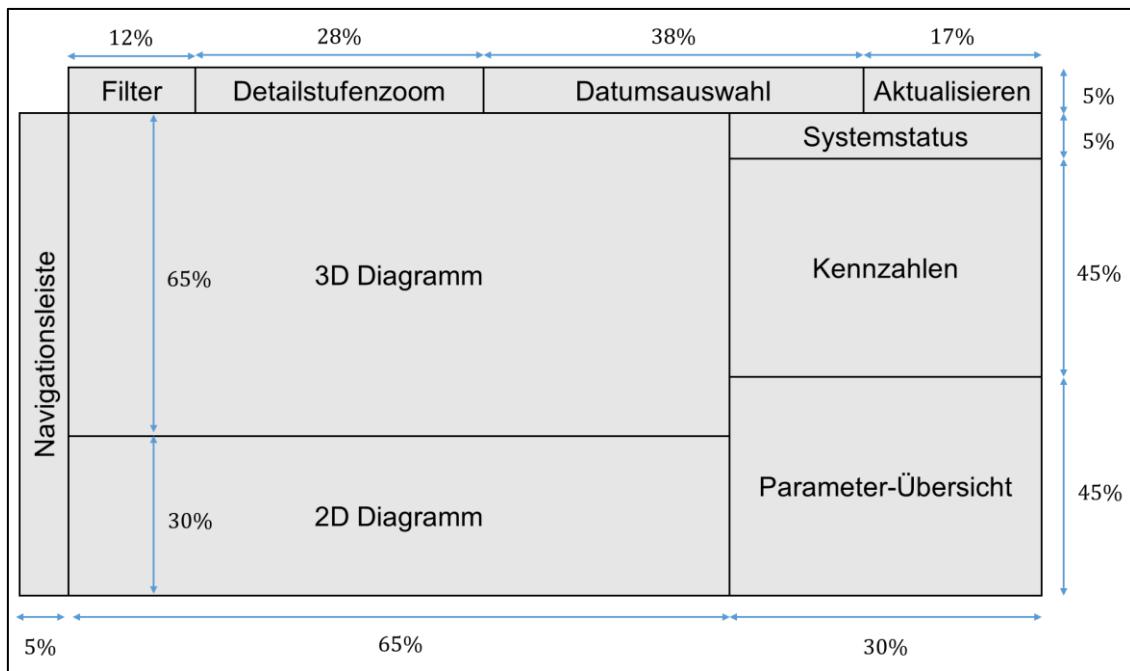


Abbildung 40: Dashboard – Größenübersicht detailliert

7.6.2 Buttons und Icons

Die in den Mockups verwendeten Buttons sind jeweils mit einem schwarzen Rahmen umgeben und haben einen leichten Schatten. Die durchgängige Verwendung dieser Eigenschaften erleichtert es dem Benutzer zu erkennen, welche Bereiche er anklicken kann.

Bei der Konzeption wurde darauf geachtet, nur Icons zu verwenden, die den meisten Benutzern bekannt sind und somit intuitiv verwendet werden können. Unerfahrene Benutzer können sich die Funktionsweise der Buttons mit einem Mouseover (oder bei Touchdisplays mit langem Druck) anzeigen lassen. Da das System voraussichtlich nur von einer geringen Anzahl an Benutzern verwendet wird, können sich diese schnell an das System gewöhnen und die Icons effizient nutzen. Die Icons in den Mockups stammen aus der Microsoft Office Piktogramme-Sammlung.

In der folgenden Tabelle sind alle in den Mockups verwendeten Icons dargestellt und kurz beschrieben (Tabelle 14).

Icon	Beschreibung	Icon	Beschreibung
	Wechselt Ansicht auf Dashboard		Wechselt Ansicht auf Parameter verwalten
	Wechselt Ansicht auf Bericht erstellen		Wechselt Ansicht auf Datenquelle ändern
	Öffnet Einstellungen		Klappt die linke Seitenleiste aus, zeigt Texte zu den Icons
	Klappt die linke Seitenleiste ein, blendet Texte zu den Icons aus		Zeigt aktive Filter an
	Öffnet Dialog, um die Detailstufe zu ändern		Öffnet Menü, um Datum auszuwählen
	Aktualisiert die aktuelle Ansicht		Öffnet das Panel im Vollbild
	Reduziert das Panel vom Vollbild in die ursprüngliche Größe		Öffnet Detaileinstellungen-Seitenleiste
	Klappt das Panel aus		Klappt das Panel ein
	Öffnet Seitenleiste, um den Diagrammtyp zu ändern (2D)		Element löschen
	Element bearbeiten		Öffnet Farbauswahldialog
	Einstellungen speichern		Gespeicherte Einstellungen laden
	Aktivieren / einblenden		Filterbedingungen umkehren
	Deaktivieren / ausblenden		Öffnet Dialog mit Panel-spezifischen Buttons

Tabelle 14: Icons

7.6.3 Schriftgrößen und -arten

In den Mockups wurde wie auch bei Kibana die Schriftart ‚Inter‘ in den unten dargestellten Schriftgrößen verwendet, da diese speziell für Computerbildschirme entwickelt wurde (Tabelle 15) [Inte2019].

Schriftgröße	Verwendung	Beispiel
16	<ul style="list-style-type: none"> • Schelleinstellungen • Linke Seitenleiste • Panelüberschriften • Buttons für Speichern, Abbrechen und Zurücksetzen 	ABCabc
12	<ul style="list-style-type: none"> • Sonstige Buttons • Zeitstempel • Allgemeine und Detaileinstellungen 	ABCabc
11	<ul style="list-style-type: none"> • Tabellen im Dashboard • Filterleiste 	ABCabc
9	<ul style="list-style-type: none"> • Achsenbeschriftungen von Diagrammen und Legenden 	ABCabc

Tabelle 15: Schriftgrößen

7.6.4 Farben

Die folgenden Zwischenüberschriften beschreiben, wo in den Mockups Farben eingesetzt wurden. In den Texten darunter wird jeweils erläutert, warum welche Farbe bzw. Farbpalette verwendet wurde. In den Tabellen werden die Farben sowie deren Farbwerten in HEX, RGB und CMYK dargestellt.

Parameter

Farbe	Kritikalität	HEX	RGB	CMYK
	Unkritisch	#fffffb2	255,255,178	0,0,30,0
	Eher unkritisch	#feccc5c	254,204,92	0,20,60,0
	Neutral	#fd8d3c	253,141,60	0,45,70,0
	Eher kritisch	#f03b20	240,59,32	5,77,80,0
	Kritisch	#bd0026	189,0,38	25,100,70,0

[Colo2019]

Tabelle 16: Farben – Parameter

Für den Farbcode bei der Färbung von Parametern wurde eine auf der Farbe Rot basierende Farbpalette verwendet. Da es sich um sequentielle Daten handelt, wurde eine sequentielle Farbpalette gewählt. Die Abstufungen der gewählten Palette sind auch für Farbenblinde problemlos unterscheidbar (Tabelle 16).

Balken der 3D-Ansicht

Falls die Balken der 3D-Ansicht nach Clustern / Rechenzentren gruppiert gefärbt werden sollen, müssen abwechselnd unterschiedliche Farben verwendet werden. Dazu wurde die untenstehende qualitative Farbpalette gewählt. Diese besteht aus zwölf gut voneinander unterscheidbaren Farben (Tabelle 17).

Farbe	HEX	RGB	CMYK
	#a6cee3	166,206,227	35,7,0,0
	#1f78b4	31,120,180	90,30,0,0
	#b2df8a	178,223,138	30,0,45,0
	#33a02c	51,160,44	80,0,100,0
	#fb9a99	251,154,153	0,40,25,0
	#e31a1c	227,26,28	10,90,80,0
	#fdbf6f	253,191,111	0,25,50,0
	#ff7f00	255,127,0	0,50,100,0
	#cab2d6	202,178,214	20,25,0,0
	#6a3d9a	106,61,154	60,70,0,0
	#ffff99	255,255,153	0,0,40,0
	#b15928	177,89,40	23,73,98,12

[Colo2019]

Tabelle 17: Farben – Balken der 3D-Ansicht

Sonstige Farben (Panels, Buttons und Hintergründe)

Abgesehen von den zuvor beschriebenen kräftigen Farben wurden für die Mockups ausschließlich die Farben Schwarz und Weiß sowie Grautöne verwendet. Dadurch entsteht ein höherer Kontrast zwischen den Elementen der Oberfläche und der Benutzer wird nicht von den Kerninhalten abgelenkt.

8 Prototyp des neuen Systems

Um die Funktionalität des Systems zu demonstrieren und für die Benutzer anschaulicher zu gestalten, wurde aus allen Mockups ein klickbarer Prototyp erstellt. Dieser liegt als PDF¹ vor, um systemunabhängig getestet werden zu können und besteht aus einer Sammlung an Mockups. Die einzelnen Schaltflächen der Mockups sind jeweils auf entsprechende Seiten im PDF verlinkt, sodass der Benutzer testen kann, welcher Button welche Funktion erfüllt. Als Beispiel ist in untenstehender Abbildung mit blauen Kästen hervorgehoben, welche Schaltflächen des Dashboards verlinkt und somit klickbar sind (Abbildung 41).



Abbildung 41: Prototyp

Es wird empfohlen, das PDF-Dokument im Vollbild zu öffnen. Da es sich nur um einen Prototypen und kein fertiges Produkt handelt, kann es bei der Bedienung zu Einschränkungen kommen. Dies kann besonders dann auftreten, wenn von verlinkten Seiten weitere Buttons angeklickt werden. Mit einem Klick auf die Fläche oben links kehrt man immer zum Dashboard zurück. Von dort sind alle Schaltflächen zuverlässig verlinkt.

¹ Link zum PDF: <https://drive.google.com/drive/folders/1d2QQv6yUAz4ydgRfRb4q7CL8xEbW7qXI>

9 Evaluation

9.1 Erfüllung der Anforderungen

Das konzipierte System soll anhand des Prototypen evaluiert werden. Dazu wird begründet, inwiefern der Prototyp die in Kapitel 5.4 geforderten Anforderungen erfüllt. Zusätzlich bewerten je zwei Benutzer der beiden Kooperationspartner den Erfüllungsgrad der einzelnen Anforderungen. Dazu bewerten sie alle Anforderungen mit Null bis Zwei Punkten (0 = Anforderung nicht erfüllt, 1 = Anforderung teilweise erfüllt, 2 = Anforderung vollständig erfüllt). Anforderungen, die durch den Prototypen nicht abgedeckt und erst bei der technischen Umsetzung erfüllt werden können, werden auf Basis der Beschreibungen in den vorherigen Kapitel bewertet.

Aus den Punktzahlen wird das arithmetische Mittel gebildet. Die Ergebnisse der Evaluation sind in untenstehender Tabelle dargestellt (Tabelle 18).

Nr.	Anforderung	Erfüllt durch	Grad der Erfüllung
1. Kernanforderungen			
1.1 Fehler und Anomalien bemerken			
1.1.1	Das System muss dem Benutzer anzeigen, wenn ein Fehler oder eine Anomalie vorliegt	Panel "Systemstatus"	2,00
1.1.2.	Das System muss den Benutzer visuell unterstützen, Fehler und Anomalien zu bemerken	Alle Panels	2,00
1.2 Fehler und Anomalien klassifizieren			
1.2.1	Das System muss dem Benutzer anzeigen, welche Art von Fehler oder Anomalie vorliegt	Panel "Systemstatus"	2,00
1.2.2	Das System muss den Benutzer visuell unterstützen zu klassifizieren, welche Art von Fehler oder Anomalie vorliegt	Panel "Systemstatus", Tabellen	1,50
1.3 Fehler und Anomalien lokalisieren			
1.3.1	Das System muss dem Benutzer anzeigen, wo ein Fehler oder eine Anomalie vorliegt.	Panel "Systemstatus"	2,00

Nr.	Anforderung	Erfüllt durch	Grad der Erfüllung
1.3.2	Das System muss den Benutzer visuell unterstützen zu lokalisieren, wo ein Fehler oder eine Anomalie vorliegt	Alle Panels	1,75
1.4	Alle relevanten Informationen in einer Ansicht bereitstellen	3D-Ansicht	2,00
1.5	Einfache Handhabung	Oberfläche orientiert an nutzerfreundlichen Tools	1,75
1.6	Kosten	Erweiterung bestehender Tools ist wirtschaftlicher, als Eigenentwicklung. Es müssen ggf. Lizenzen gekauft werden	1,00
1.7	Unterstütze Datenquellen	Diverse Datenquellen voreingestellt, zusätzliche möglich	2,00
1.8	Einfache Anpassbarkeit von Dashboard und Reports	Report und Dashboard auf Basis verfügbarer Einstellungen frei konfigurierbar, komplexe Anpassungen nicht von normalem Benutzer durchführbar	1,50
1.9	Qualität der Visualisierungen	Diagramme orientieren sich an denen von nutzerfreundlichen Tools	1,75
2. Funktionalität			
2.1	Das System muss eine intuitive, moderne und gut strukturierte Benutzeroberfläche besitzen	Oberfläche orientiert sich an nutzerfreundlichen Tools	1,75
2.2	Das Erstellen von Dashboards muss realisierbar sein	Dashboard ist standardmäßig vorhanden. Das Erstellen von zusätzlichen Dashboards ist nicht vorgesehen	2,00
2.3	Ein Alerting beim Überschreiten von Grenzwerten muss eingerichtet werden können	Alerting kann im Panel ‚Systemstatus‘ eingerichtet werden	2,00
2.4	Die Auswertung von historischen Daten muss möglich sein	Datenquellen unterstützen historische und live Daten	2,00
2.5	Das Durchsuchen von Logdaten muss durchführbar sein	Erstellung von Logdateien ist nicht vorgesehen	0,00
2.6	Das System muss Analytics Funktionen wie bspw. Machine Learning unterstützen oder Schnittstellen zu ‚R‘ oder Python anbieten	Panel ‚Systemstatus‘	2,00
2.7	Unterstützung der in Tabelle 8 aufgelisteten sowie Möglichkeit zur Verwendung eigener Parameter	Ansicht ‚Parameter verwalten‘	2,00
2.8	„Go live“-Schalter	Datumsauswahl in den Schnelleinstellungen	2,00

Nr.	Anforderung	Erfüllt durch	Grad der Erfüllung
2.9	Datum-Slider für 3D-Diagramm	In Panel "3D-Ansicht" vorhanden	2,00
2.10	Zeitstempel mit Wochentag	In Panel "3D-Ansicht" vorhanden	2,00
2.11	Erstellung eines Reports	In Ansicht "Bericht erstellen" vorhanden	2,00
3. Usability			
3.1	Gib Benutzern die Kontrolle		
3.1.1	Nutze Modi mit Bedacht (gleiche Buttons = gleiche Aktionen)	Siehe Kapitel 7.6.2	2,00
3.1.2	Ermögliche die Bedienung jeweils mit Tastatur und Maus	Siehe Kapitel 7.5.1	1,75
3.1.3	Ermögliche Benutzern den Fokus zu ändern	Siehe Kapitel 7.5.4	2,00
3.1.4	Zeige beschreibende Nachrichten und Texte an	Texte vorhanden, Nachrichten müssen bei Umsetzung eingerichtet werden	1,25
3.1.5	Ermögliche sofortige und reversible Aktionen	Aktionen in Hauptansicht werden sofort wirksam, Aktionen in Seitenleiste erst nach Speichern	2,00
3.1.6	Sinnvolle Navigation mit Abbruchmöglichkeiten	Flexibles Wechseln zwischen Ansichten möglich, Seitenleisten enthalten Abbruch-Button	2,00
3.1.7	Berücksichtige Benutzer verschiedener Qualifikationsstufen	Erfahrene Benutzer können Dialoge verkürzen	2,00
3.1.8	Machen Sie die Benutzeroberfläche transparent (der Benutzer sollte sich jederzeit voll auf seine Aufgabe fokussieren können)	Nicht benötigte Buttons können ausgeblendet werden, es sind nur die jeweils benötigten Einstellungen sichtbar	2,00
3.1.9	Ermögliche Benutzern das Anpassen der Benutzeroberfläche	Panels können neu angeordnet werden. Grundlegende Änderungen an der Oberfläche durch den Benutzer sind nicht vorgesehen	1,00
3.1.10	Ermögliche Benutzern das direkte Bearbeiten von Objekten der Oberfläche	Diagramme sind interaktiv, alle Panels besitzen detailliertere Einstellungen	2,00
3.2	Reduziere den Gedächtnisaufwand der Benutzer		
3.2.1	Entlaste das Kurzzeitgedächtnis	Einstellungen in der Seitenleiste werden als Vorschau direkt aktiv, sodass der Benutzer sich nicht merken muss, was bereits eingestellt wurde	1,50

Nr.	Anforderung	Erfüllt durch	Grad der Erfüllung
3.2.2	Unterstütze den Benutzer dabei Elemente wiederzuerkennen (z. B. durch Tooltips oder kontextabhängige Hinweise)	Tooltips vorhanden	1,75
3.2.3	Stelle visuelle Hinweise bereit	Ausgewählte Balken oder Tabellenzellen werden farbig hervorgehoben, Buttons besitzen Tooltips	2,00
3.2.4	Ermögliche Standardeinstellungen sowie Aktionen rückgängig zu machen und zu wiederholen	Standardeinstellungen und Optionen für rückgängig machen und zurücksetzen sind vorhanden	2,00
3.2.5	Stelle Shortcuts bereit	Werden von den zu erweitern den Systemen übernommen	Derzeit nicht bewertbar
3.2.6	Strebe eine objektorientierte Syntax an	Muss bei der Umsetzung berücksichtigt werden	Derzeit nicht bewertbar
3.2.7	Verwende reale Metaphern	Siehe Kapitel 7.6.2	2,00
3.2.8	Nutze progressive Offenlegung von Funktionen (zeige dem Benutzer Funktionen nur dann, wenn er sie benötigt)	Einstellungen sind in eine separate Seitenleiste ausgelagert	2,00
3.2.9	Strebe visuelle Klarheit an	Einheitlicher Aufbau von Panels, Ansichten und Menüs	1,75
3.3	Strebe Konsistenz an		
3.3.1	Erhalte den Kontext der Aufgaben der Benutzer (Kontinuität)	Das Menü und der Schnelleinstellungen-Bereich sind in allen Ansichten sichtbar, der Benutzer weiß somit immer wo er ist und wie er zurück gelangen kann	2,00
3.3.2	Stelle Konsistenz zu anderen Systemen her (ähnlicher Aufbau von Menüs und gleiche Funktionsweise von bekannten Buttons)	Oberfläche orientiert an nutzerfreundlichen Tools	2,00
3.3.3	Gleiche Aktionen sollten zu gleichen Ergebnissen führen	Siehe Kapitel 7.6.2	2,00
3.3.4	Strebe ein optisch ansprechendes Design an	Oberfläche orientiert an nutzerfreundlichen Tools	1,75
3.3.5	Ermutige den Benutzer Funktionen auszuprobieren	3D-Ansicht ist interaktiv, der Benutzer kann Einstellungen ausprobieren ohne "etwas zu verstehen"	2,00
Mittelwert der Punktzahlen			1,82

Tabelle 18: Evaluation

9.2 Verbesserungspotenziale

Im Folgenden wird auf die Anforderungen eingegangen, die weniger als zwei Punkte erreicht haben. Auf Basis des Feedbacks der Kooperationspartner wird erläutert, wo Verbesserungspotenziale liegen, bzw. begründet, warum manche Anforderungen nicht vollständig erfüllt wurden (Tabelle 19).

Nr.	Anforderung	Verbesserungspotenzial / Kommentar
1.2.2 und 1.3.2	Das System muss den Benutzer visuell unterstützen zu klassifizieren und zu lokalisieren, welche Art von Fehler oder Anomalie vorliegt	<p>Die tatsächliche Klassifikation von Fehlern oder Anomalien wird nur im Systemstatus Panel angezeigt. In den anderen Ansichten kann aus Auffälligkeiten geschlossen werden, um welche Art es sich handelt, eine konkrete Anzeige dort ist nicht vorhanden.</p> <p>Es ist jedoch vorgesehen, dass Meldungen aus dem Systemstatus Panel ausgewählt werden können, wodurch die entsprechenden Parameter und Zeiten automatisch in den Diagrammen angezeigt werden können.</p>
1.5	Einfache Handhabung	Da es sich um einen Prototypen handelt, funktionieren nicht alle Buttons wie gewünscht.
1.6	Kosten	Es gilt im individuellen Fall zu prüfen, welches der bestehenden Tools auf Basis der eigenen Bedürfnisse am besten erweitert werden kann oder ob eine Eigenentwicklung sinnvoller ist. Erst daraus ergeben sich die konkreten Kosten.
1.8	Einfach Anpassbarkeit von Dashboard und Reports	Die Elemente des Dashboards können nach eigenen Bedürfnissen neu angeordnet werden. Darüber hinaus können die Panels sowie der Report über die entsprechende Seitenleiste im Detail konfiguriert werden. Dabei steht eine feste Auswahl an Optionen zur Verfügung, die für die meisten Benutzer ausreichen wird. Falls komplexere Anpassungen gewünscht sind, müssen diese entsprechend von Programmierern umgesetzt werden.
1.9	Qualität der Visualisierungen	Die derzeitigen Visualisierungen basieren auf Testdaten und sollen lediglich veranschaulichen, welche Arten von Diagrammen im finalen Tool eingesetzt werden sollen. Die Qualität der Visualisierungen basiert im Wesentlichen darauf, welches Tool verwendet wird.
2.1	Das System muss eine intuitive, moderne und gut strukturierte Benutzeroberfläche besitzen	Die Struktur der Oberfläche wird durch das Konzept vorgegeben. Das Zusammenspiel zwischen einzelnen Elementen sollte auf Basis des finalen Tools erneut mit den Benutzern evaluiert werden.
2.5	Das Durchsuchen von Logdaten muss durchführbar sein	Für interne und externe Kommunikation kann ein PDF-Report erstellt werden; getroffene Einstellungen und Anpassungen am Dashboard können zurückgesetzt werden. Daher wird die Erstellung von Logdateien nicht benötigt und ist somit nicht Teil der Konzeption. Es ist jedoch nicht auszuschließen, dass das genutzte Tool diese Funktion unterstützt.

Nr.	Anforderung	Verbesserungspotenzial / Kommentar
3.1.2	Ermögliche die Bedienung jeweils mit Tastatur und Maus	Es können nahezu alle Funktionen jeweils mit Tastatur oder Maus verwendet werden. Lediglich bei der Erstellung von benutzerdefinierten Labels muss zur Texteingabe mit der Maus eine virtuelle Tastatur herangezogen werden.
3.1.4	Zeige beschreibende Nachrichten und Texte an	Sämtliche Einstellungen sind ausführlich beschrieben, Buttons zeigen Tooltips an. Bestätigungsdialoge müssen bei der Umsetzung eingerichtet werden.
3.1.9	Ermögliche Benutzern das Anpassen der Benutzeroberfläche	Siehe Kommentar zu Anforderung 1.8.
3.2.1	Entlaste das Kurzzeitgedächtnis	Der Benutzer sieht alles, was aktuell benötigt wird, Einstellungen in der Seitenleiste werden als Vorschau direkt aktiv, sodass der Benutzer sich nicht merken muss was bereits eingestellt wurde.
3.2.2	Unterstütze den Benutzer dabei Elemente wiederzuerkennen (z. B. durch Tooltips oder kontextabhängige Hinweise)	Tooltips und kontextabhängige Hinweise sind im Konzept vorgesehen. Diese müssen technisch jedoch so umgesetzt werden, dass sie den Benutzer bestmöglich unterstützen.
3.2.9	Strebe visuelle Klarheit an	Das Konzept sieht einheitliche Panels, Ansichten und Menüs vor. Bei der technischen Umsetzung können vorgefertigte Elemente verwendet werden, welche die Einheitlichkeit zusätzlich unterstützen.
3.3.4	Strebe ein optisch ansprechendes Design an	Siehe Kommentar zu Anforderung 2.1.

Tabelle 19: Verbesserungspotenziale

9.3 Fazit

Das arithmetische Mittel aller bewerteten Anforderungen liegt bei 1,82 Punkten und erreicht damit rund 91% der maximalen Punktzahl (Tabelle 18). Insgesamt wurde der Prototyp also von den vier Benutzern als sehr gut bewertet.

Einschränkend gilt zu sagen, dass die Evaluation am Prototypen durchgeführt wurde und einige Funktionen nur auf Basis von Beschreibungen bewertet wurden. Daher sollte zukünftig nach der technischen Umsetzung eine weitere Evaluation erfolgen, um sicherzustellen, dass der vollständige Funktionsumfang ebenfalls den Anforderungen der Benutzer genügt.

10 Zusammenfassung und Diskussion

In diesem Kapitel wird die Arbeit chronologisch zusammengefasst und es werden die verwendeten Methoden und Werkzeuge sowie das Ergebnis diskutiert.

10.1 Zusammenfassung

Es galt folgende Forschungsfragen zu beantworten:

1. Welche Tools existieren im Umfeld von IT-Operations Analytics?
2. Inwiefern sind existierende Tools im Umfeld von IT-Operations Analytics gegenüber heutigen Anforderungen unzureichend?
3. Wie lässt sich die Lücke zwischen existierenden Tools im Umfeld von IT-Operations Analytics und heutigen Anforderungen aufheben?

Zu Beginn des vierphasigen Vorgehensmodells wurde der Ist-Zustand ermittelt. Dabei wurde untersucht, welche Tools im Umfeld von IT-Operations Analytics existieren (Frage 1). Bei der Recherche wurden zwölf Tools näher betrachtet.

Anschließend wurden Anforderungen gesammelt, die das im Rahmen dieser Arbeit konzipierte System erfüllen soll. Ermittelt wurden die Anforderungen durch eine Kombination aus Experteninterviews und Brainstorming mit den Benutzern und Literaturrecherche. Aus letzterer wurden ein Whitepaper mit ITOA-Anforderungen und eine Auflistung von Usability-Kriterien verwendet. Die einzelnen Anforderungen wurden schließlich zu einer umfassenden Anforderungsliste aggregiert.

In einem Zwischenfazit wurde festgehalten, dass acht der zwölf betrachteten Tools auf die Überwachung von Rechenzentren ausgelegt sind, von denen es jedoch keines ermöglicht, die gesamte Serverlandschaft auf einen Blick darzustellen. Nur vier der Tools bieten genügend Flexibilität, die Oberfläche nach eigenen Wünschen zu verändern. Für die vier Tools existiert jedoch kein Konzept, wie eine Oberfläche zur Überwachung von Rechenzentren gestaltet werden soll (Frage 2). Daraus resultiert die Notwendigkeit der Erstellung eines solchen. Weiterhin wurde begründet, dass eine 3D-Darstellung in das System aufgenommen werden soll, da diese im Gegensatz

zu den Ansichten der betrachteten Tools eklatante Vorteile bietet: Eine 3D-Visualisierung ermöglicht es, alle Server, Cluster und Rechenzentren auf einen Blick darzustellen. Neben der Möglichkeit, mehrere Parameter in einer Ansicht zu bündeln, kann die Anordnung der 3D-Balken auch dazu genutzt werden, um Zusammenhänge zwischen Clustern/Rechenzentren zu visualisieren. Abschließend wurde gesagt, dass ein prototypisches System mit einer 3D-Darstellung als Kernelement konzipiert werden soll, um die identifizierten Lücken aufzuheben (Frage 3).

In der Designphase wurde festgelegt, dass das System aus fünf Ansichten bestehen soll, wobei das Dashboard als zentrale Ansicht fünf Panels enthalten wird. Diese Panels enthalten die dreidimensionale Darstellung sowie zusätzlich ein zweidimensionales Diagramm, eine ML-basiertes Systemstatus-Panel und Tabellen zu Kennzahlen und als Parameter-Übersicht. Neben dem Dashboard existieren Ansichten, um die verfügbaren Parameter zu verwalten, einen PDF-Bericht zu erstellen, die Datenquelle zu ändern und für allgemeine Einstellungen. All diese Elemente wurden in Form von Mockups konzipiert und anhand derer umfassend beschrieben.

Aus allen Mockups wurde ein klickbarer Prototyp erstellt. Dieser demonstriert die Funktionalität des Systems und wurde für die Evaluation mit den Benutzern herangezogen.

Bei der Evaluation wurde einerseits begründet, inwiefern der Prototyp die geforderten Anforderungen erfüllt und andererseits haben jeweils zwei Benutzer der beiden Kooperationspartner den Prototypen getestet. Dabei haben sie anhand des Prototypen und auf Basis der Funktionsbeschreibungen bewertet, inwiefern die geforderten Anforderungen erfüllt wurden. Einige wenige Anforderungen konnten nicht bewertet werden, da sie Teil der technischen Umsetzung sind. Insgesamt hat der Prototyp bei der Bewertung 91% der maximal möglichen Punktzahl erhalten.

Abschließend kann festgehalten werden, dass der Prototyp als sehr gut bewertet wurde und als Grundlage für die technische Umsetzung genutzt werden kann. Da diese nicht Teil der vorliegenden Arbeit sind, werden im Ausblick Hinweise gegeben, was bei der technischen Umsetzung beachtet werden sollte (Kapitel 11.1).

10.2 Diskussion

In diesem Kapitel werden die verwendeten Methoden und Werkzeuge bezüglich ihres Nutzen für die Arbeit sowie die Ergebnisse allgemein bewertet.

Konzipiertes System

Bei dem erstellten Prototyp handelt es sich um kein finales System, wodurch nicht festgestellt werden kann, wie hoch die Akzeptanz im späteren Einsatz ausfallen wird. Insgesamt erfüllt der Prototyp nahezu alle geforderten Anforderungen in hohem Maße. Aus Gründen der Veranschaulichung hat sich die Methode des Prototyping sehr gut geeignet. Auch war die Verwendung von Microsoft PowerPoint für die Erstellung von Mockups und Prototyp absolut ausreichend. Im Vergleich mit alternativen Tools konnte PowerPoint mit allen benötigten Funktionen mithalten. Zusätzlich könnten die Mockups von Dritten leicht angepasst werden, da PowerPoint eine hohe Verbreitung besitzt und viele Benutzer damit umgehen können.

Vorgehensmodell

Insgesamt stellte sich das agile Vorgehensmodell für die Konzeption des Systems als sehr zielführend heraus. Insbesondere die enge Zusammenarbeit mit den Benutzern hat dazu geführt, dass das System sehr nah an deren Bedürfnissen und auf sie zugeschnitten konzipiert werden konnte. Die Analysephase hat wesentliche Grundlagen hervorgebracht und zu einer umfassenden Analyse des Ist-Zustandes geführt. Die Phasen ‚Gestaltung‘ und ‚Konstruktion‘ haben ebenfalls stark von der Benutzerzentrierung profitiert. Hier konnten Zwischenergebnisse direkt präsentiert und auf Basis des Feedbacks angepasst werden. In der finalen Evaluation wurden die Benutzer dann noch einmal um Feedback gebeten, was schließlich mit der insgesamt sehr guten Bewertung des Prototypen endete.

Die Hauptschwäche des Vorgehensmodells liegt darin, dass in der Konstruktionsphase lediglich ein Prototyp und kein fertiges System erstellt wurde. Dadurch konnten die Benutzer nicht alle Funktionen testen und auch nicht mit ihren eigenen Daten arbeiten. Daher wurde versucht, die Funktionsbeschreibungen möglichst umfangreich zu gestalten, um die fehlende Funktionalität dadurch zu kompensieren.

Als weitere Schwäche ist zu nennen, dass nur eine kleine Gruppe an Benutzern befragt wurde und die positiven Ergebnisse somit nicht zwangsläufig ebenso für andere Benutzer gelten müssen. Da sich die Bewertungen der Benutzer der Kooperationspartner jedoch nur minimal unterschieden haben, kann diese Schwäche vernachlässigt werden. In jedem Fall sollte das finale System mit den zukünftigen Benutzern umfassend erneut evaluiert werden.

Literaturrecherche

Die Literaturrecherche eignete sich sehr gut, um den Ist-Zustand zu erfassen. Es konnten sowohl aktuelle Tools als auch Anforderungen aus dem Bereich der Wissenschaft gesammelt werden. Einschränkend kann gesagt werden, dass im Bereich der aktuellen Tools keine vollständige Marktanalyse durchgeführt wurde, sondern dass eine ausgewählte Sammlung von verbreiteten Tools analysiert wurde. Der Fokus dieser Arbeit sollte stärker auf der Gestaltung und Konstruktion des Systems liegen, daher wurde sich auf die Rechercheergebnisse von Chircu et al. verlassen [CSB+2019].

Experteninterviews und Brainstorming

Die Kommunikation mit den Benutzern als zentrales Element des Vorgehensmodells und insbesondere die Kombination aus Experteninterviews und Brainstorming hat maßgeblich zum Erfolg der Konzeption beigetragen. Durch die Interviews konnten der Ist-Zustand bei den Kooperationspartnern sowie grundlegende Anforderungen erhoben werden. Beim Brainstorming konnten diese dann näher spezifiziert werden und zusammen mit Mockups und Prototypen wurden iterativ Verbesserungen vorgenommen.

11 Ausblick

In diesem Kapitel werden Hinweise zur technischen Umsetzung des Systems gegeben sowie Grenzen und mögliche Erweiterungen aufgezeigt.

11.1 Technische Umsetzung

Im Folgenden werden zuerst allgemeine und anschließend konkrete Hinweise für die Tools der Kooperationspartner gegeben.

11.1.1 Allgemeine Hinweise

Bei der technischen Umsetzung geht es im Wesentlichen darum, den Prototypen auf Basis der Funktionsbeschreibungen und unter Berücksichtigung sämtlicher Anforderungen in ein nutzbares System zu überführen. Dabei sollten die Benutzer analog zum bisherigen Vorgehensmodell stark eingebunden werden, um deren Akzeptanz sicherzustellen.

Grundsätzlich gilt es abzuwägen, ob das System von Grund auf neu programmiert wird oder ob ein vorhandenes Tool genutzt und um das vorliegende Konzept erweitert wird. Diese Entscheidung muss im individuellen Fall abgewogen werden. Unter Berücksichtigung des enormen Aufwandes einer Neuprogrammierung empfiehlt es sich jedoch eines der mächtigen vorhandenen Tools – wie Kibana oder Grafana – zu erweitern, um so von Vorteilen wie eines technischen Supports oder einer aktiven Community profitieren zu können. Diese Vorteile gilt es ebenfalls im individuellen Fall gegen die anfallenden Kosten abzuwägen.

Das Konzept sieht vor, dass alle Elemente auf dem Bildschirm sichtbar sind, ohne dass der Benutzer scrollen muss. Das finale System muss sich also auf verschiedene Bildschirmgrößen skalieren lassen. Dabei gilt es zu berücksichtigen, dass die Vorgabe, ab einer minimalen Größe keine Scrollbalken zu verwenden, nicht mehr eingehalten werden kann. Dabei müssen die Elemente des Dashboards und der Einstellungen dann automatisch untereinander angeordnet werden.

Die konkreten Designelemente des Prototypen müssen nicht eins zu eins übernommen werden. Vielmehr bietet es sich an, Elemente von vorhandenen Tools einzusetzen und / oder das Design an die unternehmenseigenen Designvorgaben anzupassen. Dazu zählen sowohl Schriftgrößen und -farben als auch Icons.

11.1.2 Hinweise für die Tools der Kooperationspartner

Die beiden Kooperationspartner nutzen bereits mächtige Tools, die grundsätzlich in der Lage sind, die geforderten Anforderungen zu erfüllen. Dazu muss lediglich die Oberfläche auf Basis des vorliegenden Konzeptes angepasst werden. Dies ist überwiegend mit bereits vorhandenen voreingestellten Funktionen möglich.

Um das Kernelement, die 3D-Darstellung, zu integrieren, können sowohl für Kibana als auch für Grafana Plugins geschrieben bzw. verwendet werden. Diese stellen dann die dreidimensionalen Balken auf Basis der angebundenen Datenquelle dar und können als Visualisierungsschicht in den jeweiligen Tools verwendet werden. Für Kibana kann dazu die Vega-Grammatik [Vega2019] und für Grafana kann Plotly eingesetzt werden [GrPl2019].

Beide Kooperationspartner besitzen eine unterschiedliche Hierarchie unterhalb der Datencenter. Die Mockups und der Prototyp wurden anhand der Daten der BA erstellt (Datencenter > Cluster > Server). Analog müssten bei der technischen Umsetzung an den entsprechenden Stellen die Hierarchiestufen des Automobilherstellers hinterlegt werden.

11.2 Grenzen

Im Folgenden werden zwei Limitierungen des vorliegenden Konzeptes erläutert. Diese beziehen sich zum einen auf die fachliche Konzeption, also die Konzeption ohne technische Umsetzung, und zum anderen auf die Anzahl gleichzeitig dargestellter Parameter.

11.2.1 Fachliches Konzept

Bei dieser Arbeit handelt es sich um ein Konzept ohne technische Umsetzung. Daher konnte die Evaluation nur auf Basis von Prototyp und Funktionsbeschreibungen durchgeführt werden. Es kann also nicht mit Sicherheit gesagt werden, dass das Konzept nach der technischen Umsetzung ein ebenso gutes Evaluationsergebnis erzielen wird. Umso wichtiger ist es, die Benutzer während der Umsetzung weiterhin zu integrieren und deren Feedback zu berücksichtigen.

11.2.2 Anzahl gleichzeitiger Parameter

Die 3D-Ansicht wurde so konzipiert, dass zwei Parameter gleichzeitig dargestellt werden (durch Höhe der Balken und Farocode). Theoretisch können jedoch noch weitere Parameter gleichzeitig dargestellt werden, beispielsweise durch die Diagrammachsen oder die Balkendicken (vgl. Kapitel 11.3.2). Darauf wurde im vorliegenden Konzept jedoch aus Gründen der Übersichtlichkeit verzichtet. Stattdessen werden die Balken entlang der Achsen auf Basis von Cluster- und Rechenzentrums-grenzen angeordnet.

11.3 Mögliche Erweiterungen

Im Folgenden werden verschiedene Optionen genannt, wie das System über das Konzept hinausgehend erweitert werden kann. Diese Erweiterungen können direkt bei der individuellen Einführung oder als Erweiterung gehandhabt werden.

11.3.1 Mobiler Einsatz des Systems

Dem allgemeinen Trend hin zu verstärktem mobilem Arbeiten [Quir2017] folgend, kann das konzipierte System auch auf mobilen Endgeräten mit ausreichend großem Display – wie zum Beispiel auf Tablets – genutzt werden. Das Konzept sieht vor, dass sämtliche Funktionen auch auf Touchdisplay-Geräten nutzbar sind. Derzeit ist der Einsatz von entsprechenden Geräten bei den Kooperationspartnern jedoch nicht vorgesehen. Daher wird diese Funktion aktuell nicht benötigt und verbleibt als mögliche Erweiterung.

11.3.2 Anzahl gleichzeitiger Parameter

Die Anzahl an gleichzeitig dargestellten Parametern in der 3D-Ansicht kann von zwei auf sieben erhöht werden. Dazu kann je ein Parameter auf eines der folgenden Elemente abgebildet werden:

- Höhe der Balken (Z-Achse)
- Farbton
- Farbsättigung
- Position auf der X-Achse
- Position auf der Y-Achse
- Breite der Balken
- Höhe der Balken

Da der erhöhte Informationsgehalt jedoch in einem schlechten Verhältnis zur reduzierten Übersichtlichkeit steht, wurden diese Optionen im vorliegenden Konzept nicht berücksichtigt. Eine größere Anzahl an Parametern sollte bestenfalls nur bei einer geringeren Anzahl an Balken eingesetzt werden.

11.3.3 Dark Mode

In dunklen Räumen finden es viele Benutzer angenehmer, auf dunklere Displays zu blicken. Dazu werden Schriften in Weiß und Hintergründe in Schwarz dargestellt [Brei2019]. Viele Tools unterstützen standardmäßig einen Dark Mode und erlauben es den Benutzern, jederzeit den Modus zu wechseln [Graf2019, Kiba2019]. Aus Komplexitätsgründen wurde der Prototyp nur in einem Modus konzipiert. Bei der Erweiterung von vorhandenen Tools, die einen Dark Mode unterstützen, oder bei einer Eigenentwicklung kann in den Einstellungen des Systems ein Schalter eingefügt werden, um zwischen den Modi zu wechseln.

Literaturverzeichnis

- [Acro2019a] Acrolinx (2019): Über das Unternehmen. <https://www.acrolinx.de/uber-acrolinx> [abgerufen am: 01.08.2019].
- [Acro2019b] Acrolinx (2019): Server Monitor Dashboard <https://docs.acrolinx.com/coreplatform/latest/de/analytics/acrolinx-analytics-dashboards/server-monitor-dashboard> [abgerufen am: 01.08.2019].
- [ArPo2004] Arregoces, Mauricio; Portolani, Maurizio (2004): Data Center Fundamentals – Understanding Data Center network design and infrastructure architecture, including load balancing, SSL, and security. Ciscopress. CCIE No. 3285. <http://estigia.fi-b.unam.mx/maestria/Cisco%20Press%20-%20Data%20Center%20Fundamentals.pdf> [abgerufen am: 01.08.2019].
- [Beck2016] Becker, Franziska (2016): Wireframes, Prototypes, Mockups – Wo liegt eigentlich der Unterschied? <https://blog.comwrap.com/wireframes-prototypes-mockups-wo-liegt-eigentlich-der-unterschied> [abgerufen am: 01.08.2019].
- [Bedf2014] Bedford, Aurora (2014): Icon Usability. usability.ch – the swiss usability center <https://www.usability.ch/news/icon-usability.html> [abgerufen am: 01.08.2019].
- [Brei2019] Breithut, Jörg (2019): Display im Dunkelmodus – Was der Dark Mode wirklich bringt. Spiegel Online. <https://www.spiegel.de/netzwelt/gadgets/display-im-dark-mode-was-der-dunkelmodus-wirklich-bringt-a-1248550.html> [abgerufen am: 01.08.2019].
- [Cham2018] Chamoni, Prof. Dr. Peter (2018): <https://www.gabler-banklexikon.de/definition/dashboard-70726/version-337043> [abgerufen am: 01.08.2019].
- [Chri2015] Christiane (2015): 7 vorgeschriebene Gesetze des User Interface Designs. 99designs. <https://99designs.de/blog/design-kreativitaet/user-interface-design/> [abgerufen am: 01.08.2019].

- [Clar1972] Clark, C. H. (1972): Brainstorming. Methoden der Zusammenarbeit und Ideenfindung, 4. Auflage, München.
- [Colo2019] Color Brewer (2019): Color Brewer 2.0 – Color Advice For Cartography. <http://colorbrewer2.org> [abgerufen am: 01.08.2019].
- [CSB+2019] Chircu, A.; Sultanow, E.; Baum, D.; Koch, C.; and Seßler, M. (2019): Visualization and Machine Learning for Data Center Management. In Proceedings of “Workshops der Informatik 2019”, 23. – 26. 09. 2019, Kassel. Springer.
- [DeSc2013] Deml, Stefan; Schwarz, Thorsten (2013): IT Operations Analytics – Definition, Einsatzbereiche, Marktübersicht. <https://amasol.de/download/whitepaper-it-operations-analytics.html> [abgerufen am: 01.08.2019].
- [DHPW2008] Dalia, Apurva F.; Harry, Craig Allen; Pacholec, Michael A. und Whitney, Robert Thompson (2008): Address Book Clearing House, United States Patent. Patentnummer 7337448. <http://www.freepatentsonline.com/7337448.html> [abgerufen am: 01.08.2019].
- [Digi2018] Digitalisierungsindex (2018): Studien 2018. <https://www.digitalisierungsindex.de/studie-2018/> [abgerufen am: 01.08.2019].
- [DrOs2017] Drilling, Thomas; Ostler, Ulrike (2017): Verbundstrukturen im Rechenzentrum – Was ist ein Cluster? <https://www.datacenter-insider.de/was-ist-ein-cluster-a-588715/> [abgerufen am: 01.08.2019].
- [Elas2019] Elasticsearch (2019): Elasticsearch – Das Kernstück des Elastic Stack. <https://www.elastic.co/de/products/elasticsearch>
- [ElKe2019] Elasticsearch Keyboard Shortcuts (2019): Kibana User Guide – Elasticsearch Keyboard shortcuts. <https://www.elastic.co/guide/en/kibana/current/keyboard-shortcuts.html> [abgerufen am: 01.08.2019].
- [Empr2019] Emprechtinger, Franz (2019): Prototyping: 9 erfolgreiche Methoden und Best Practices. Lead – Innovation Management. <https://www.lead-innovation.com/blog/prototyping-methoden-und-best-practices> [abgerufen am: 01.08.2019].

- [Extr2019] ExtraHop (2019): ExtraHop – Rise Above The Noise <https://www.extrahop.com> [abgerufen am: 01.08.2019].
- [FKK+1995] Flick, Uwe; Kardorff, Ernst von; Keupp, Heiner; Rosenstiel, Lutz von und Wolff, Stephan (1995): Handbuch qualitative Sozialforschung. Grundlagen, Konzepte, Methoden und Anwendungen. Weinheim (Beltz, Psychologie-Verlag-Union).
- [Garr2010] Garrett, Jesse (2010): The Elements of User Experience: User-Centered Design for the Web and Beyond (Voices That Matter).
- [GLa2010] Gläser, Jochen; Laudel, Grit (2010): Experteninterviews und Qualitative Inhaltsanalyse. Wiesbaden (VS Verlag).
- [Götz2016] Götze, Marcel (2016): Usability and User-Interface-Design. Human-Computer-Interaction User-Interface-Design. Universität Magdeburg. <http://www.uni-magdeburg.de/fgse/dekanat/lehre/wiki/lib/exe/fetch.php/lehre/wise2016/lmnm/hci.pdf> [abgerufen am: 01.08.2019].
- [Graf2019] Grafana (2019): Grafana Labs. <https://grafana.com/> [abgerufen am: 01.08.2019].
- [GrKe2019] Grafana Keyboard Shortcuts (2019): Grafana Labs – Grafana Keyboard Shortcuts. <https://grafana.com/docs/features/shortcuts/> [abgerufen am: 01.08.2019].
- [GrPl2019] Grafana Plotly (2019): Plotly Panel for Grafana. <https://grafana.com/grafana/plugins/natel-plotly-panel> [abgerufen am: 01.08.2019].
- [Hens2015] Henschel, Angelina (2015): Konzept und Design einer grafischen Benutzeroberfläche eines datenzentrierten Prozess-Management-Systems. Universität Ulm. <http://dbis.eprints.uni-ulm.de/1360/> [abgerufen am: 01.08.2019].
- [IDC2015] IDC (2015): HP Operations Analytics Simplifies and Accelerates IT Service Delivery and Digital Innovation, S.5&8. Zitiert nach [Maya2016].

- [Inte2019] Inter (2019): The Inter typeface family. <https://rsms.me/inter/> [abgerufen am: 01.08.2019].
- [ISO2012] BS EN ISO 9241-143 (2012): Ergonomie der Mensch-System-Interaktion – Teil 143: Formulardialoge (ISO 9241-143:2012).
- [ISO2016] DIN EN ISO 9241-11 (2016): Ergonomie der Mensch-System-Interaktion – Teil 11: Gebrauchstauglichkeit: Begriffe und Konzepte (ISO/DIS 9241-11.2:2016).
- [Kamp2018] Kamps, Prof. Dr. Udo (2018): Zeitreihenanalyse. Gabler Wirtschaftslexikon. <https://wirtschaftslexikon.gabler.de/definition/zeitreihenanalyse-48908/> [abgerufen am: 01.08.2019].
- [Kiba2019] Kibana-Elasticsearch (2019): Kibana – Ihr Fenster in den Elastic Stack. <https://www.elastic.co/de/products/kibana>
- [KiDe2019] Kibana Demo (2019): Kibana Demo Dashboards. https://demo.elastic.co/app/kibana#/dashboards?_g=0
- [Korn2016] Kornmeier, Martin (2016): Wissenschaftlich schreiben leicht gemacht. Für Bachelor, Master und Dissertation. Paderborn, München (UTB).
- [Lamn2016] Lamnek, Siegfried (2016): Qualitative Sozialforschung. Lehrbuch. Langensalza (Beltz).
- [Like2015] LikeMag (2015): 8% aller Männer und 0.5% aller Frauen sind farbenblind. <https://likemag.com/de/8-aller-maenner-und-0-5-alter-frauen-sind-farbenblind-und-du-der-test-sagt-dir-die-wahrheit/15224> [abgerufen am: 01.08.2019].
- [Mand2012] Mandel, Theo (2012): The Elements of User Interface Design. <https://theomandel.com/wp-content/uploads/2012/07/Mandel-GoldenRules.pdf> [abgerufen am: 01.08.2019].

- [Maya2016] Mayato - Heise, Timo (2016): IT-Operations Analytics (ITOA) - Den IT-Betrieb mit Big-Data Analytics optimieren. <https://www.mayato.com/wp-content/uploads/2016/12/WP-White-Paper-IT-Operations-Analytics-ITOA-IT-Betrieb-Big-Data-Analytics.pdf> [abgerufen am: 01.08.2019].
- [Micr2019] Microsoft (2019): Windows Admin Center. <https://www.microsoft.com/de-de/cloud-platform/windows-admin-center> [abgerufen am: 01.08.2019].
- [Muni2019] Munin (2019): Munin Überwachungstool. <http://munin-monitoring.org> [abgerufen am: 01.08.2019].
- [Nagi2019] Nagios (2019): Nagios – The Industry Standard in IT Infrastructure Monitoring. Nagios Network Analyzer - Netflow Analysis, Monitoring, and Bandwidth Utilization Software. <https://www.nagios.com/> [abgerufen am: 01.08.2019].
- [Niel1993] Nielsen, Jakob (1993): Usability Engineering. Stanford (Elsevier Science). Academic Press.
- [Orac2019] Oracle (2019): Oracle Cloud – Operational Intelligence for Modern IT. <https://cloud.oracle.com> [abgerufen am: 01.08.2019].
- [OsMi2016] Ostler, Ulrike; Mirko (2016): Definition Server – Warum heißt der Server Server? DataCenter Insider. <https://www.datacenter-insider.de/warum-heisst-der-server-server-a-567972/> [abgerufen am: 01.08.2019].
- [Pern2017] Pernice, Kara (2017): Don’t “Validate” Designs; Test Them. Nielsen Norman Group – Worls Leaders in Research-Based User Experience. <https://www.nngroup.com/articles/no-validate-in-ux/> [abgerufen am: 01.08.2019].
- [Pous2018] Pousschi, Key (2018): Vorlesungsskript "Mobile and Digital Business", Universität Potsdam.

- [Quir2017] Quirin, Iris (2017): Mobiles Arbeiten: Die neuen Digitalnomaden. Digitaler Mittelstand. <https://digitaler-mittelstand.de/trends/ratgeber/mobiles-arbeiten-die-neuen-digitalnomaden-40229> [abgerufen am: 01.07.2019].
- [RaAu2017] Rauch, Gedeon; Augsten, Stephan (2017): Definition „Graphical User Interface“ – Was ist eine GUI? <https://www.dev-insider.de/was-ist-eine-gui-a-651868/> [abgerufen am: 01.08.2019].
- [Riew2010] Riewerts, Christoph (2010): Vorlesungsskript DHBW Stuttgart, Informatik, SW-Engineering, Kapitel 4.3. © Prof. Dr. Helmut Balzert V2.0. <http://wwwlehre.dhbw-stuttgart.de/~rie/Vorlesungsskripte/DesignErgoVorlesung.pdf> [abgerufen am: 01.08.2019].
- [Royc1970] Royce, Winston W. (1970): „Managing the Development of Large Software Systems: Concepts and Techniques“, In: Technical Papers of Western Electronic Show and Convention (WesCon) August 25-28, 1970, Los Angeles, USA.
- [Rüdi2019] Rüdiger, Adriane (2019): Fast lückenlos verfügbar – Was ist Hochverfügbarkeit. DataCenter Insider. <https://www.datacenter-insider.de/was-ist-hochverfuegbarkeit-a-821602/> [abgerufen am: 01.08.2019].
- [Rusu2017] Rusu, Viorel (2017): 3D Charts Visualizations for Kibana. Universität Rey Juan Carlos, Spanien. https://virusu.github.io/3D_kibana_charts_vis/ [abgerufen am: 01.08.2019].
- [Ryte2014] Ryte Wiki (2014): Serverauslastung. <https://de.ryte.com/wiki/Serverauslastung> [abgerufen am: 01.08.2019].
- [Self2017] selfhtml (2017): CSS/Tutorials/Tooltips mit CSS https://wiki.selfhtml.org/wiki/CSS/Tutorials/Tooltips_mit_CSS [abgerufen am: 01.08.2019].

- [Somm2015] Sommergut, Wolfgang (2015): Windows Pro – Windows, Virtualisierung und Cloud für Profis – Server -Auslastung überwachen mit dem kostenlosen Windows Capacity Check. <https://www.windowspro.de/tool/server-auslastung-ueberwachen-dem-kostenlosen-windows-capacity-check> [abgerufen am: 01.08.2019].
- [Splu2019] Splunk (2019): Splunk – Erhalten Sie Antworten aus Maschinen-daten <https://www.splunk.com/de/de> [abgerufen am: 01.08.2019].
- [TaBM2007] Taylor, T.; Brooks, S. und McHugh, J. (2007): NetBytes Viewer: An Entity-based NetFlow Visualization Utility for Identifying Intrusive Behavior. In Proceedings of the Workshop on Visualization for Computer Security (VizSEC 2007).
- [True2019] TrueSight bmc (2019): bmc Software – TrueSight – Umsetzung und Neuerfindung von IT-Prozessen und Cloud-Operationen zur Steige-rung der Unternehmensagilität. <http://www.bmcsoftware.de/it-solutions/truesight.html> [abgerufen am: 01.08.2019].
- [Vega2019] Vega (2019): Vega – A Visualization Grammar. <https://vega.github.io/vega> [abgerufen am: 01.08.2019].
- [Wert1923] Wertheimer, Max (1923): Untersuchungen zur Lehre von der Gestalt. In: Psychologische Forschung. Zeitschrift für Psychologie und ihre Grenzwis-senschaften. Bd. 4. Koffka, K. [u.a.] (Hrsg.). Berlin: Springer.
- [Wiki2019] Wikipedia (2019): Prototyping (Softwareentwicklung). [https://de.wikipedia.org/w/index.php?title=Prototyping_\(Softwareentwicklung\)&oldid=189506910](https://de.wikipedia.org/w/index.php?title=Prototyping_(Softwareentwicklung)&oldid=189506910) [abgerufen am: 01.08.2019].
- [WuBu2015] Wu, Caesar; Buyya, Rajkumar (2015): Cloud Data Centers and Cost Modeling – A Complete Guide to Planning, Designing and Building a Cloud Data Center. <https://doi.org/10.1016/C2013-0-23202-5>.
- [Zitz2018] Zitzer, Bernhard (2018): Lohnt sich die Sidebar heute noch? Ja oder Nein – Eine Abwägung. Xovi Expertenrat. <https://www.xovi.de/loht-sich-die-sidebar-noch-ja-oder-nein> [abgerufen am: 01.08.2019].

Anhang

Anforderungen aus ITOA – Sicherheit und Architektur

Nr.	Anforderung
4.	Sicherheit
4.1	Eine revisionssichere Speicherung von Logdaten muss möglich sein
4.2	Die Passwörter müssen verschlüsselt werden
4.3	Die Kommunikation zwischen Quellsystem und ITOA-System muss verschlüsselt sein
4.4	Die Kommunikation zwischen dem Anwender des ITOA-Systems und dem ITOA-Systems selbst muss verschlüsselt sein
4.5	Ein rollenbasiertes Berechtigungskonzept muss umsetzbar sein
5.	Architektur
5.1	Aktuelle Plattformen wie Windows, Linux und Mac werden unterstützt
5.2	Der Import von Logdaten / die Anbindung von Logdaten-Quellen muss möglich sein
5.3	Die Entwicklung von Erweiterungen muss realisierbar sein
5.4	Die Architektur muss stark skalierbar sein und Big Data-Analytics ermöglichen
5.5	Die Architektur muss Auswertungen in Echtzeit durchführen können
5.6	Die Architektur muss gegen einen unberechtigten Zugriff von innen und außen sicher geschützt sein

[IDC2015]

Tabelle 20: Anforderungen aus ITOA – Sicherheit und Architektur

Eidesstattliche Erklärung

Ich erkläre hiermit, dass ich diese Masterarbeit selbstständig ohne Hilfe Dritter und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Alle den benutzten Quellen wörtlich oder sinngemäß entnommenen Stellen sind als solche einzeln kenntlich gemacht.

Diese Arbeit ist bislang keiner anderen Prüfungsbehörde vorgelegt worden und auch nicht veröffentlicht worden.

Ich bin mir bewusst, dass eine falsche Erklärung rechtliche Folgen haben wird.

Potsdam, 09.08.2019



Ort, Datum, Unterschrift