

Inhalt

Geleitwort zur 1. Auflage.	XI
---------------------------------	----

Überblick: Dokumentationsmittel im Buch.	XIII
---	------

1 Warum Softwarearchitekturen dokumentieren?	1
1.1 Montagmorgen	1
1.1.1 Fragen über Fragen	1
1.1.2 Wer fragt, bekommt Antworten	2
1.2 Voll unagil?	5
1.2.1 Agil vorgehen	5
1.2.2 Funktionierende Software vor umfassender Dokumentation	6
1.2.3 Dokumentation unterstützt Kommunikation.	7
1.3 Wirkungsvolle Architekturdokumentation	7
1.3.1 Ziel 1: Architekturarbeit unterstützen	8
1.3.2 Ziel 2: Architektur nachvollziehbar und bewertbar machen	8
1.3.3 Ziel 3: Umsetzung und Weiterentwicklung leiten.	9
1.3.4 Fremdwort Do ku men ta tion [...zion] [lat.]	9
1.4 Mission Statement für dieses Buch.	10
1.5 Über dieses Buch	11
1.5.1 Für wen ich dieses Buch geschrieben habe.	11
1.5.2 Wie dieses Buch aufgebaut ist.	12
1.5.3 Wem ich danke schön sagen möchte	16
2 Was Softwarearchitektur ist und worauf sie aufbaut	17
2.1 Softwarearchitektur-Freischwimmer	17
2.1.1 Was ist Softwarearchitektur?	17
2.1.2 Wie entsteht Softwarearchitektur?	18
2.1.3 Softwarearchitekt/in (m/w/d) gesucht.	21
2.1.4 Ein Architekturüberblick auf n Seiten, n < 30.	23
2.2 Die Zielsetzung vermitteln.	23
2.2.1 Jetzt kommt ein Karton!	23
2.2.2 Virtueller Produktkarton (Dokumentationsmittel)	24
2.2.3 Fallbeispiel: Schach-Engine „DokChess“	26

2.2.4	Tipps zum Erstellen von Produktkartons.	27
2.2.5	Fallbeispiel: Schachplattform „immer-nur-schach.de“	27
2.3	Den Kontext abgrenzen	28
2.3.1	Systemkontext (Dokumentationsmittel).	29
2.3.2	Fallbeispiel: Systemkontext „immer-nur-schach.de“	31
2.3.3	Tipps zur Erstellung des Systemkontextes	32
2.4	Im Rahmen bleiben.	37
2.4.1	Warum Randbedingungen festhalten?	37
2.4.2	Randbedingungen (Dokumentationsmittel).	38
2.4.3	Fallbeispiel: Randbedingungen „immer-nur-schach.de“	39
2.4.4	Tipps zum Festhalten von Randbedingungen	40
2.5	Geforderte Qualitätsmerkmale	42
2.5.1	Was sind Qualitätsmerkmale?	42
2.5.2	Qualitätsziele (Dokumentationsmittel)	44
2.5.3	Fallbeispiel: Qualitätsziele „immer-nur-schach.de“	45
2.5.4	Fallbeispiel: Qualitätsziele „DokChess“	45
2.5.5	Qualitätsmerkmale genauer beschreiben	46
2.5.6	Qualitätsszenarien (Dokumentationsmittel)	47
2.5.7	Fallbeispiel: Qualitätsszenarien „immer-nur-schach.de“	49
2.5.8	Tipps zum Festhalten von Qualitätsszenarien	51
2.6	Weitere Einflüsse und Hilfsmittel	53
2.6.1	Stakeholder	54
2.6.2	Persona (Dokumentationsmittel)	55
2.6.3	Fallbeispiel: Persona „immer-nur-schach.de“	57
2.6.4	Risiken	58
2.6.5	Technische Risiken (Dokumentationsmittel).	58
2.6.6	Fallbeispiel: Technische Risiken „DokChess“	60
2.6.7	Glossar (Dokumentationsmittel)	60
3	Entscheidungen treffen und festhalten.	63
3.1	Historisch gewachsen?	63
3.2	Architekturentscheidungen	64
3.2.1	Architekturentscheidung (Dokumentationsmittel).	64
3.2.2	Fallbeispiel: Spannende Fragen „DokChess“	67
3.2.3	Tipps zur Formulierung von Fragestellungen	67
3.2.4	Fallbeispiel: Fragestellungen „immer-nur-schach.de“	69
3.2.5	ADRs als eine alternative Strukturierung	72
3.3	Einflussfaktoren auf Entscheidungen.	73
3.3.1	Den Überblick behalten	73
3.3.2	Kreuztabellen	74
3.3.3	Fallbeispiel: Einflüsse „immer-nur-schach.de“	75
3.3.4	Tipps zur Anfertigung von Kreuztabellen	75
3.4	Kompakte Darstellung der Lösungsstrategie	77

3.4.1	Softwarearchitektur auf einem Bierdeckel?	77
3.4.2	Lösungsstrategie (Dokumentationsmittel)	77
3.4.3	Fallbeispiel: Lösungsstrategie „DokChess“	80
3.4.4	Als Ergänzung: ein Überblicksbild	81
3.4.5	Eine Architekturbewertung auf dem Bierdeckel	81
4	Plädoyer für eine feste Gliederung	83
4.1	Aus Essener Feder	83
4.2	Vorteile einer festen Struktur	85
4.3	arc42 – Vorschlag für eine Gliederung	87
4.3.1	Was ist arc42?	87
4.3.2	Die Struktur der arc42-Vorlage	88
4.3.3	Wo funktioniert arc42 besonders gut?	90
4.3.4	arc42 in diesem Buch	90
4.4	Alternativen zu arc42	91
4.4.1	Standards zur Architekturbeschreibung	92
4.4.2	Vorgehensmodelle	93
4.4.3	Architektur-Frameworks	95
4.4.4	Fachliteratur als Inspiration?	96
5	Sichten auf Softwarearchitektur	101
5.1	Strukturen entwerfen und festhalten	101
5.1.1	Was ist was? Band 127: Unser Softwaresystem	101
5.1.2	Schritte der Zerlegung dokumentieren	102
5.1.3	Bausteinsicht (Dokumentationsmittel)	103
5.1.4	Fallbeispiel: Bausteinsicht „DokChess“ (Ausschnitt)	106
5.1.5	Komponenten: Babylonische Sprachverwirrung 2.0.	106
5.1.6	Tipps zur Erstellung der Bausteinsicht	108
5.1.7	Interaktionspunkte beschreiben	111
5.1.8	Schnittstellenbeschreibung (Dokumentationsmittel)	115
5.1.9	Fallbeispiel: Schnittstellen der Eröffnung in „DokChess“	117
5.2	Verschiedene Blickwinkel	119
5.2.1	Hat Mozart modelliert?	119
5.2.2	Fachliche Zerlegung vs. technische Zerlegung	121
5.2.3	Fallbeispiel: Bausteinsicht „immer-nur-schach.de“	123
5.3	Verhalten und Abläufe beschreiben	126
5.3.1	Abläufe in Entwurf und Dokumentation	126
5.3.2	Darstellungen für Abläufe	126
5.3.3	Laufzeitsicht (Dokumentationsmittel)	129
5.3.4	Fallbeispiel: Ablauf in DokChess	130
5.3.5	Fallbeispiel: Zustandsautomat XBoard (DokChess)	130
5.4	Die Dinge zum Einsatz bringen	131
5.4.1	Betriebsaspekte in der Architekturdokumentation	132

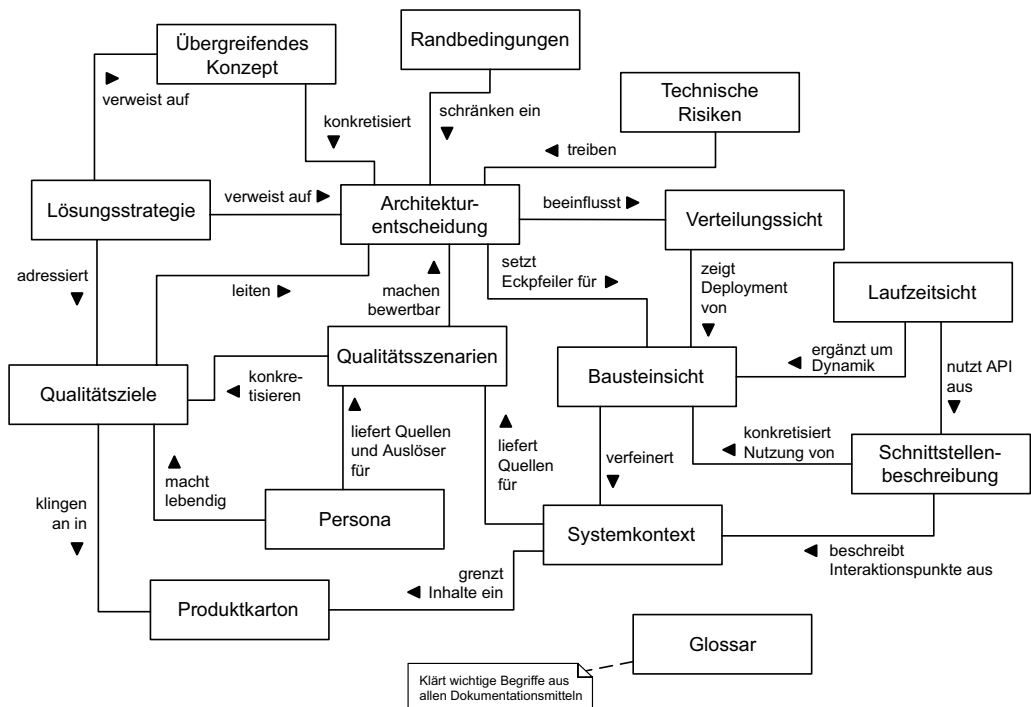
5.4.2	Darstellungen für Verteilung.....	133
5.4.3	Verteilungssicht (Dokumentationsmittel)	135
5.4.4	Fallbeispiel: „immer-nur-schach.de“.....	137
5.5	Alternative Vorschläge für Sichten	138
5.6	Muster kommunizieren	142
5.6.1	Muster in der Softwareentwicklung.....	142
5.6.2	Wann sollten Sie Muster dokumentieren (und wo)?.....	142
5.6.3	Einsatz von Mustern dokumentieren	143
5.6.4	Fallbeispiel: DokChess	145
6	Übergreifende Konzepte	147
6.1	Warum übergreifende Themen?	147
6.2	Themen und Lösungsoptionen.	149
6.2.1	Mögliche Themen für übergreifende Konzepte.....	150
6.2.2	Typische Lösungsoptionen	151
6.3	Themenauswahl	153
6.3.1	Wie wählen Sie Themen für die Dokumentation aus?	153
6.3.2	Fallbeispiel: Übergreifende Themen „DokChess“.....	155
6.4	Eine Gliederungstechnik für Konzepte.....	156
6.4.1	Werkzeug: Warum? Was? Wie? Wohin noch?.....	157
6.4.2	Gliederung für ein Konzept.....	159
6.4.3	Informeller Text für den Architekturüberblick	161
6.5	Tipps zur Erstellung übergreifender Konzepte	162
7	Werkzeuge zur Dokumentation	165
7.1	Notationen passgenau wählen.....	165
7.2	Toolparade zur Architekturdokumentation	170
7.2.1	Erstellung und Pflege.....	170
7.2.2	Verwaltung von Inhalten.....	177
7.2.3	Kommunikation von Lösungen.....	180
7.3	Repository: UML vs. Wiki.....	182
7.3.1	Steht alles im Wiki?	182
7.3.2	Steht alles im UML-Tool?.....	186
7.3.3	UML-Tool + Wiki == Traumpaar?	189
7.4	Docs-as-Code als Trend.....	190
7.5	Wie auswählen?.....	193
8	Lightfäden für das Vorgehen zur Dokumentation	195
8.1	Während der Entwicklung dokumentieren	195
8.1.1	Zielgruppen Ihrer Dokumentation	196
8.1.2	Dokumentationsmittel und Dokumente.....	198
8.1.3	Womit anfangen?	201
8.1.4	Während der Arbeit: Kommunizieren und Pflegen	202

8.2	Der Softwaredetektiv: Bestehendes dokumentieren	204
8.2.1	Auslöser für Dokumentationsbedarf	205
8.2.2	Mögliche Szenarien und Ziele des Dokumentierens im Nachhinein	205
8.2.3	Sherlock Holmes vs. Die drei ???	206
8.2.4	Informationsquellen identifizieren	207
8.2.5	Dokumentationsmittel unter der Lupe	209
8.2.6	Exkurs: Werkzeuge zur Rekonstruktion	213
8.3	Variationen von „Ein System“	218
8.3.1	Dokumentation von Systemlandschaften	219
8.3.2	Dokumentation von Frameworks und Blue Prints	221
8.3.3	Große, heterogene Systeme und Microservices-Lösungen	222
8.4	Standpunkt: Mein minimaler Architekturüberblick	225
9	Architekturüberblick DokChess	227
9.1	Einführung und Ziele	228
9.1.1	Aufgabenstellung	228
9.1.2	Qualitätsziele	228
9.1.3	Stakeholder	229
9.2	Randbedingungen	232
9.2.1	Technische Randbedingungen	232
9.2.2	Organisatorische Randbedingungen	232
9.2.3	Konventionen	233
9.3	Kontextabgrenzung	234
9.3.1	Fachlicher Kontext	234
9.3.2	Technischer Kontext oder Verteilungskontext	235
9.4	Lösungsstrategie	236
9.4.1	Aufbau von DokChess	237
9.4.2	Spielstrategie	238
9.4.3	Die Anbindung	238
9.5	Bausteinsicht	239
9.5.1	Ebene 1: Gesamtsystem (Whitebox)	239
9.5.2	XBoard-Protokoll (Blackbox)	240
9.5.3	Spielregeln (Blackbox)	241
9.5.4	Engine (Blackbox)	242
9.5.5	Eröffnung (Blackbox)	243
9.5.6	Ebene 2: Engine (Whitebox)	245
9.5.7	Zugsuche (Blackbox)	245
9.5.8	Stellungsbewertung (Blackbox)	247
9.6	Laufzeitsicht	248
9.6.1	Zugermittlung Walkthrough	248
9.7	Verteilungssicht	249
9.7.1	Infrastruktur Windows	249
9.8	Querschnittliche Konzepte	251

9.8.1	Abhängigkeiten zwischen Modulen	251
9.8.2	Schach-Domänenmodell.....	251
9.8.3	Benutzungsoberfläche	253
9.8.4	Plausibilisierung und Validierung	254
9.8.5	Ausnahme- und Fehlerbehandlung.....	255
9.8.6	Logging, Protokollierung, Tracing	255
9.8.7	Testbarkeit.....	256
9.9	Entwurfsentscheidungen	258
9.9.1	Wie kommuniziert die Engine mit der Außenwelt?	258
9.9.2	Sind Stellungsobjekte veränderlich oder nicht?	259
9.10	Qualitätsanforderungen	262
9.10.1	Qualitätsbaum.....	262
9.10.2	Qualitätsszenarien	263
9.11	Risiken und technische Schulden	265
9.11.1	Risiko: Anbindung an das Frontend schlägt fehl	265
9.11.2	Risiko: Aufwand der Implementierung zu hoch	265
9.11.3	Risiko: Erreichen der Spielstärke scheitert	266
9.12	Glossar	267
10	Stolpersteine der Architekturdokumentation	269
10.1	Probleme	269
10.2	Fiese Fallen	271
10.3	... und wie man sie umgeht oder entschärft.....	273
10.4	Reviews von Architekturdokumentation	275
	Glossar	283
	Literaturverzeichnis.....	287
	Stichwortverzeichnis.....	291

Überblick: Dokumentationsmittel im Buch

Die Abbildung zeigt alle im Buch vorgestellten Dokumentationsmittel („Zutaten“) für Softwarearchitektur. Verbindungslinien visualisieren wichtige methodische Zusammenhänge. Die Pfeile an den Linien geben die Leserichtung für die Beschriftung an (Beispiel: Bausteinsicht verfeinert Systemkontext).



Dokumentationsmittel des Buchs mit wichtigen Zusammenhängen

Der Tabelle auf der nächsten Seite können Sie entnehmen, in welchem Abschnitt im Buch Sie den Steckbrief zum betreffenden Dokumentationsmittel finden.

Überblick über die Dokumentationsmittel

Dokumentationsmittel	Nutzen	Steckbrief
Architekturentscheidung	Nachvollziehbare Darstellung einer zentralen, risikoreichen Entscheidung	Abschnitt 3.2.1
Bausteinsicht	Visualisierung der Struktur des Softwaresystems und wie die Teile voneinander abhängen	Abschnitt 5.1.3
Glossar	Etablieren eines einheitlichen Wortschatzes im ganzen Vorhaben	Abschnitt 2.6.7
Laufzeitsicht	Visualisierung von dynamischen Strukturen und Verhalten, vor allem von Abläufen	Abschnitt 5.3.3
Lösungsstrategie	Stark verdichteter Architekturüberblick; Gegenüberstellung der wichtigsten Ziele und Lösungsansätze	Abschnitt 3.4.2
Persona	Archetypische Beschreibung einer Personen-gruppe und deren Ziele (Stakeholder)	Abschnitt 2.6.2
Produktkarton	Plakative Darstellung der wesentlichen Funktionen, Ziele und Merkmale des Systems	Abschnitt 2.2.2
Qualitätsszenarien	Konkretisierung von Qualitätsanforderungen in kurzen, beispielhaften Sätzen	Abschnitt 2.5.6
Qualitätsziele	Motivation der wichtigsten an das System gestellten Qualitätsanforderungen	Abschnitt 2.5.2
Randbedingungen	Sammlung der technischen bzw. organisatori-schen Vorgaben, die beim Entwurf einzuhalten sind (oder waren)	Abschnitt 2.4.2
Schnittstellen-beschreibung	Detaillierte Beschreibung, wie ein Baustein Funk-tionalität bereitstellt (oder welche er benötigt)	Abschnitt 5.1.8
Systemkontext	Visualisierung der Fremdsysteme und Benutzer, mit denen das System interagiert	Abschnitt 2.3.1
Technische Risiken	Beschreibung der Risiken, die Einfluss auf die Softwarearchitektur haben (oder hatten)	Abschnitt 2.6.5
Übergreifendes Konzept	Darstellung eines übergreifenden Themas, zur Vereinheitlichung im System oder zur Detaillierung eines Ansatzes	Abschnitt 6.4.2
Verteilungssicht	Visualisierung der Zielumgebung, der Inbetrieb-nahme und des Betriebs des Systems	Abschnitt 5.4.3

Diese Leseprobe haben Sie beim
 edv-buchversand.de heruntergeladen.
 Das Buch können Sie online in unserem
 Shop bestellen.
[Hier zum Shop](#)