

Inhaltsverzeichnis

Abbildungsverzeichnis	v
Tabellenverzeichnis	viii
Abkürzungsverzeichnis	ix
Zusammenfassung	x
Abstract	xi
1 Einleitung und Motivation	1
1.1 Qualitätssicherungs- und Inbetriebnahmesysteme	3
1.2 Fehlerdiagnose als Aufgabe in der Instandhaltung	6
1.3 Testmanagement als Aufgabe in der Inbetriebnahmeplanung	7
1.4 Schlussfolgerung und Zielstellung der Arbeit	9
1.5 Aufbau der Arbeit nach der Design Science Research Methodology	9
2 Grundlagen und Stand der Technik: Assistenzsysteme zu Fehlerdiagnose und Testmanagement für Produktionssysteme der diskreten Fertigung	12
2.1 Digitale Assistenzsysteme in der diskreten Fertigung	12
2.2 Wissensbasierte Assistenzsysteme und Expertensysteme	12
2.2.1 Wissensrepräsentation	13
2.2.2 Ziehen von Schlussfolgerungen	16
2.2.3 Maschinelle Lernfähigkeit	18
2.2.4 Spezielle Ausprägungen von wissensbasierten Assistenzsystemen	18
2.2.5 Aufbau von wissensbasierten Systemen	19
2.3 Grundlagen der Fehlerdiagnose	21
2.3.1 Der Begriff Fehler	21
2.3.2 Ablauf einer Fehlerdiagnose	22
2.3.3 Unterstützung der Fehlerdiagnose durch Symptomerfassung	22
2.3.4 Phasen der Fehlerdiagnose	23
2.4 Methoden zur softwarebasierten Unterstützung der Fehlerdiagnose	24
2.4.1 Signalbasierte Fehlerdiagnose	25
2.4.2 Modellbasierte Fehlerdiagnose	26
2.4.3 Fehlerdiagnose mit Methoden der Klassifikation	26

2.4.4	Fehlerdiagnose mit Methoden des Schlussfolgerns	27
2.4.5	Fallbasierte Fehlerdiagnose durch Case-based Reasoning	27
2.5	Stand der Technik: Aktuelle Trends zur industriellen Fehlerdiagnose und Instandhaltung in der diskreten Fertigung	27
2.5.1	Industrie 4.0	28
2.5.2	Cyber-physische (Produktions-) Systeme (CPS/CPPS)	28
2.5.3	Predictive Maintenance	28
2.5.4	Predictive Quality	29
2.5.5	Prognostics and Health Management (PHM)	29
2.5.6	Digitaler Zwilling	29
2.5.7	Big Data und Cloud-Technologien	30
2.5.8	Edge Computing	30
2.5.9	Prozessansätze zur Qualitätssteigerung	31
2.5.10	Maschinelles Lernen und künstliche Intelligenz	32
2.5.11	Data Loop zur kontinuierlichen Entwicklung	32
2.5.12	Condition-based Monitoring (CBM)	33
2.6	Grundlagen des Testmanagements	33
2.6.1	Der Testprozess	33
2.6.2	Testfallauswahl- und Priorisierung im Regressionstest	34
2.7	Stand der Technik: Aktuelle Trends zum industriellen Testmanagement und zur Testfallpriorisierung in der diskreten Fertigung	35
2.7.1	Agentenbasiertes Testmanagement	36
2.7.2	Regressionstests bei Systemänderungen und Rekonfigurationen	36
2.8	Zusammenfassende Bewertung und Ableitung der Forschungslücke und der Forschungsfragen	37
3	Synergien durch ein gemeinsames Assistenzsystem zur Fehlerdiagnose und zum Testmanagement	39
3.1	Generisches Systemmodell	40
3.2	Der Fehlerdiagnoseprozess	41
3.2.1	Daten, Wissen und Erfahrung im Diagnoseprozess	43
3.2.2	Unterstützung des Diagnoseprozesses durch das Assistenzsystem	45
3.3	Der Testplanungs- und Testmanagementprozess	47
3.3.1	Daten, Wissen und Erfahrung im Testmanagementprozess	48
3.3.2	Unterstützung des Testmanagementprozesses durch das Assistenzsystem	50
3.4	Synergien durch wechselseitig genutzte Assistenzfunktionen	51
3.5	Synergien durch die Verwendung eines gemeinsamen Datenmodells	53
3.5.1	Fehler und Fehlersymptome als zentrale Elemente für Fehlerdiagnose und Testmanagement	53
3.5.2	Bereichsübergreifende Datenflüsse für die Assistenzfunktionen	55

4 Konzipierung des Teilsystems zur Fehlerdiagnose	59
4.1 Funktionsweise, Randbedingungen und Entwurfsentscheidungen	59
4.1.1 Nutzerrollen für die Assistenzfunktion zur Fehlerdiagnose	61
4.1.2 Erkennen eines zu diagnostizierenden Fehlers	61
4.1.3 Umfang eines Diagnosefalls	62
4.1.4 Zentraler Aufbau des Assistenzsystems	63
4.2 Ableiten von Symptomen aus der Systemüberwachung	63
4.2.1 Ergebnis der Symptomidentifikation: Der formalisierte Diagnosefall	64
4.2.2 Datenquellen zur Identifikation von Symptomen	64
4.2.3 Automatische Symptomidentifikation	67
4.2.4 Symptomidentifikation anhand von Logfiles oder Traces	67
4.2.5 Symptomidentifikation anhand der Ergebnisdatenbank	68
4.2.6 Symptomidentifikation aus archivierten Messwerten	68
4.2.7 Manuelle Symptomidentifikation durch den Systemexperten (GUI)	69
4.2.8 Eingabe von Symptomen und Regeln durch die Systemexperten	69
4.3 Datenmodell für Fehlerfallhistorie und zu diagnostizierenden Systemzustand	69
4.4 Automatisches Bestimmen der Fehlerursache	70
4.4.1 Symptombasiertes Case-based Reasoning	70
4.4.2 Retrieve: Finden der wahrscheinlichsten Fehlerursache	73
4.4.3 Reuse: Sortierte Liste bereits aufgetretener Fehler nach Übereinstimmung	75
4.4.4 Revise: Korrekturen durch den Systemexperten	76
4.4.5 Retain: Speicherung von Fällen mit ihrer Ursache	77
4.5 Auftrag an Testmanagementfunktion: Finden von passenden Testfällen, um Symptome zu provozieren	77
4.5.1 Unterscheidung von mehrdeutigem Diagnoseergebnis durch das Provozieren weiterer Symptome	78
4.5.2 Absicherung der Fehlerkorrektur durch passende Testfälle	79
5 Konzipierung des Teilsystems zum Testmanagement	81
5.1 Funktionsweise, Randbedingungen und Entwurfsentscheidungen	82
5.1.1 Nutzerrollen für die Assistenzfunktion zum Testmanagement	83
5.1.2 Iterative Weiterentwicklung als Auslöser des Systemtests	84
5.1.3 Umfang eines Systemtests	84
5.1.4 Aufstellen eines Testplans	84
5.1.5 Datenquellen	85
5.2 Umgang mit unsicherem und unscharfem Wissen	85
5.3 Metamodell zur Abbildung des Systemmodells	86
5.4 Datenmodell mit Testfällen, Systemmodell und Fehlerfallhistorie	86
5.5 Filtern von relevanten Testfällen für das Provozieren von Symptomen	88
5.6 Finden von Abdeckungslücken	89

5.7	Bestimmen des Testbedarfs von Subfunktionen und Symptomen	90
5.7.1	Änderungen	91
5.7.2	Metriken für Subfunktionen	91
5.7.3	Metriken aus der Fehlerfallhistorie	92
5.7.4	Berechnen des Testbedarfs mithilfe der Metriken	92
5.8	Bestimmen der Testfallperformance	96
5.8.1	Globale Metriken als Grundlage für die Testfallperformance	97
5.8.2	Berechnung der Testfallperformance bezogen auf Subfunktionen	97
5.8.3	Berechnung der Testfallperformance bezogen auf Symptome	98
5.9	Berechnen der resultierenden Testfallprioritäten	99
6	Evaluation des Konzepts durch Realisierung für ein QS&IBN-System der Automobilindustrie	101
6.1	Evaluationskonzept	101
6.2	Aufbau des Systems aus der Automobilindustrie	102
6.3	Proof-of-Concept: Prototypische Realisierung	104
6.3.1	Entwurfsentscheidungen und Randbedingungen	104
6.3.2	Aufbau des Assistenzsystem-Prototyps	105
6.3.3	Realisierung der Assistenzfunktion zur Fehlerdiagnose	106
6.3.4	Realisierung der Assistenzfunktion zum Testmanagement	112
6.3.5	Fazit zum Proof-of-Concept	120
6.4	Proof-of-Value: Einsatz des Prototyps am QS&IBN-System der Automobilindustrie	122
6.4.1	Systemmodellierung und Ablauf der Evaluierung	122
6.4.2	Herausforderung natürlichsprachliche Formulierung	123
6.4.3	Durchgeführte Testfälle	124
6.4.4	Ergebnis der Fehlerdiagnose	124
6.4.5	Ergebnis der Testfallpriorisierung	126
6.4.6	Fazit zum Proof-of-Value	127
6.5	Generalisierbarkeit des Konzepts	128
6.6	Betrachtung der Evaluierungsergebnisse anhand der Forschungsfragen	128
7	Schlussbetrachtung	132
7.1	Fazit der Ergebnisse	132
7.2	Ausblick auf weiterführende Forschung	134
Literaturverzeichnis		136