

**Dokumentation einer konzeptuellen Ausarbeitung eines
Condition Monitoring Systems des Moduls
Software-Architektur**

vorgelegt von

Alexander Huber

Manh-Khang Richard Mai

Hendrik Haas

betreut und begutachtet von

Prof. Dr. Markus Esch

Saarbrücken, 22. März 2020

Inhaltsverzeichnis

1	Einleitung	1
1.1	Problemstellung	1
1.2	Zielsetzung	1
1.3	Stakeholder	2
1.4	Gliederung	2
2	Einflussfaktoren	3
2.1	Organisatorische Einflussfaktoren	3
2.2	Technische Einflussfaktoren	3
3	Kontextabgrenzung	5
3.1	Fachlicher Kontext	5
3.2	Technischer Kontext	6
4	Qualitaetsszenarien	7
5	Konzept	11
5.1	Lösungsstrategie	11
5.2	Bausteinsicht	11
5.3	Laufzeitsicht	11
5.4	Verteilungssicht	11
6	Fazit	13
	Literatur	15
	Abbildungsverzeichnis	17
	Tabellenverzeichnis	17
	Listings	17
	Abkürzungsverzeichnis	19
A	Erster Abschnitt des Anhangs	23

1 Einleitung

Seit Jahrzehnten versuchen die Menschen ihre maschinellen Systeme immer effizienter und sicherer zu gestalten. Durch das Aufkommen von Computern besteht seitdem die Möglichkeit diese Systeme digital und nutzerfreundlich zu überwachen. So nehmen sie sensorische Werte der Anlage auf und verarbeiten diese. Es können so beispielsweise Abschätzungen mittels linearer Regression über die Lebensdauer einer solchen Anlage erfasst und diese dann in einem Diagramm dargestellt werden. Solche Systeme sind und werden in Zukunft noch viel wichtiger werden, da hier statistische Methoden eingesetzt werden können, die für den modernen Fertigungs- und Wartungsprozess unerlässlich sind.

1.1 Problemstellung

Die Firma HYDAC Systems und Services GmbH entwickelt derzeit ein System zum erfassen, verarbeiten und auswerten von Messwerten und Prozessdaten in Echtzeit. Allgemein ist solch ein System auch unter dem Begriff Condition Monitoring bekannt und ermöglicht es Prozesse zu überwachen, zu planen und gegebenenfalls zu optimieren. Es wird demnach zur Prozessoptimierung verwendet und erhöht die Verfügbarkeit von Anlagen. Im Rahmen der Machbarkeit und Untersuchung unterschiedlicher Herangehensweisen, ist es für die Firma interessant wie Studenten der Hochschule htwsaar ein Condition Monitoring System anhand gegebener Anforderungen konzipieren würden. Folgende Anforderungen verdeutlichen die zentrale fachliche Aufgabenstellung:

- Vordefinierte Datensätze sind aus einer Datenquelle gefiltert oder ungefiltert zu empfangen und in einer Datenbank persistent abzuspeichern. Dabei kann diese Datenquelle verschiedene Protokolle sein, wie HFI-MM, HFI-CM, Modbus TCP/RTU, CAN/CANopen, S7 protocol, OPC-UA (client and server), HTTP und WebSocket
- Das System benötigt eine Benutzerverwaltung um Nutzern und Gruppen Zugriffsrechte auf dem System zu ermöglichen. Die Zugriffsrechte sind lesend, schreibend und administrativ. Solange man administrative Rechte hat kann man die Rechte anderen Nutzern und Gruppen verteilen.
- Es wird ein Web-Interface benötigt um auf die einzelnen Komponenten zugreifen zu können. So soll es ein Dashboard geben, das individuell angepasst werden kann und beispielsweise Prozesse anzeigt.

1.2 Zielsetzung

Das Ziel dieser Arbeit soll es sein, einen Architekturentwurf für ein Condition Monitoring auszuarbeiten, welcher unter Anwendung moderner Architekturansätze zu gestalten ist. Die folgenden primären Qualitätsziele sind dabei genau zu beachten:

- Effizienz (Performance | Leistung): Das System soll ca. 500 Datensätze innerhalb einer Sekunde aus ein oder mehreren Datenquellen erfassen können.

1 Einleitung

- Verfügbarkeit (Erreichbarkeit): Das System soll eine Hochverfügbarkeit von mindestens 99,9% aufweisen und darf demnach maximal 8h 45min im Jahr ausfallen.
- Nutzbarkeit (UX): Das User-Interface soll einfach zu verstehen sein und einem Bediener innerhalb einer Stunde erlernbar sein. Des weiteren soll die Oberfläche für den Benutzer individuell anpassbar sein.

1.3 Stakeholder

Die Stakeholder werden durch eine Stakeholderanalyse identifiziert und bezeichnen die für das Projekt relevanten Personengruppen. Hierzu ist festzuhalten, welchen Vorteil die jeweiligen Gruppen haben und welche individuellen Erwartungen an das System gestellt werden. Durch eine Marktanalyse in Betrachtung des Einsatzumfelds und anhand der Einsatzmöglichkeiten haben sich uns folgende Personengruppen als Stakeholder an das System herausgestellt:

- Fach- und Führungskräfte
- Anwendungs- und Testentwickler
- Linien- und Produktionskräfte
- Systemadministratoren

Die Fach- und Führungskräfte sind insbesondere an einer zielführenden Übersicht und einer Aggregation von planbaren Artefakten interessiert, welches ihnen wiederum bei der Unterstützung und Planung des Arbeitsprozesses hilft. Die Anwendungs- und Testentwickler erwarten einen niedrigen Integrationsaufwand bei Implementierung neuer Funktionalitäten. Die Linien- und Produktionskräfte erwarten eine einfache Ansicht der Zustandserfassung, sodass sie das System einfach überwachen können. Die Systemadministratoren erwarten, dass das System eine sofortige Informationsmitteilung und eine einfache Maßnahmenkontrolle bereitstellt, damit einerseits die Systeme auf Fehler überwacht und reagiert werden können.

1.4 Gliederung

Das erste Kapitel gibt einen Überblick über die Themen, welche folgend in der Dokumentation ausformuliert sind. Es schafft Klarheit darüber wer, warum und wozu solch ein System benötigt wird und definiert die zentralen Merkmale. Im Kapitel 2, wird dann auf Randbedingungen eingegangen, die nicht direkt in den Anforderungen beschreiben sind, welche aber trotzdem bei einer Entwicklung beachtet werden sollen. Das 5 befasst sich danach mit den gestellten Anforderungen und beschreibt ein mögliches Konzept nach Kriterien. Kapitel Im letzten Kapitel 6 werden wir das Vorgehen abschließend beurteilen und einen Ausblick wagen.

2 Einflussfaktoren

2.1 Organisatorische Einflussfaktoren

2.2 Technische Einflussfaktoren

3 Kontextabgrenzung

In der Kontextabgrenzung grenzen wir alle Kommunikationspartner vom System ab und stellen somit unsere externen Schnittstellen fest. Dazu wird der Kontext einerseits fachlich, als auch technisch voneinander abgegrenzt.

3.1 Fachlicher Kontext

Die fachliche Kontextabgrenzung dient uns dazu alle Kommunikationspartner zusammen mit den jeweiligen ein und Ausgabedaten mit dem System zu erläutern. Folgende Abbildung veranschaulicht die fachliche Kontextabgrenzung:

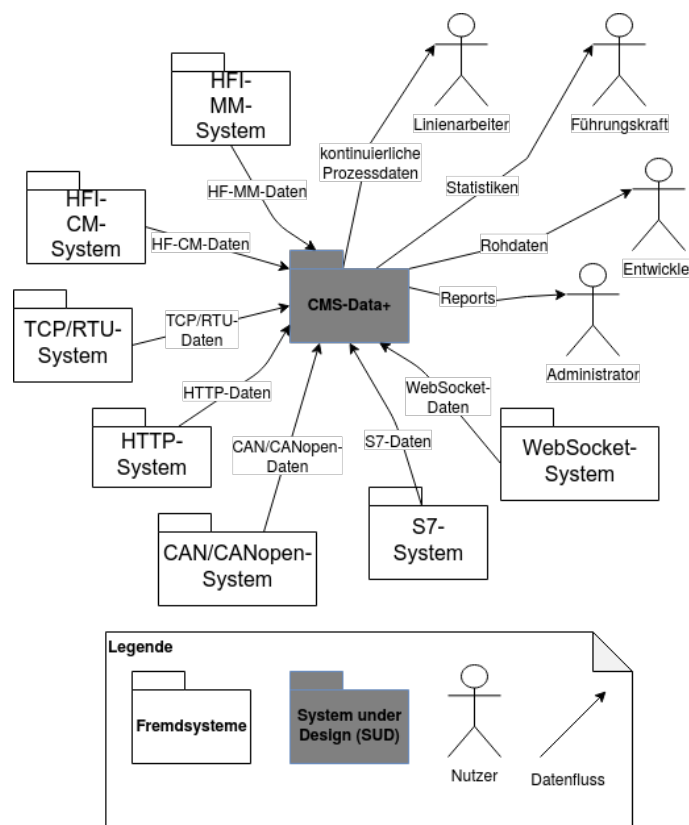


Abbildung 3.1: Fachliche Kontextabgrenzung

Wie das Diagramm veranschaulicht, dient das Data+ als Datensenke für alle angezeigten Systeme. Jedes dieser Systeme kann über eine eigene Schnittstelle bzw. Protokoll verfügen und über jenes die protokollspezifischen Daten an das Data+ senden. Folgende Tabelle stellt noch einmal die Kommunikationspartner und den fachlichen Kontext zum System dar:

3 Kontextabgrenzung

Kommunikations-partner	Eingabe	Ausgabe
HFI-MM-System		
HFI-CM-System		
TCP/RTU-System		
HTTP- System		
CAN/CANopen-System		
S7-System		
WebSocket-System		

Tabelle 3.1: Fachliche Kontextabgrenzung der Kommunikationspartner

3.2 Technischer Kontext

Der technische Kontext definiert die Kanäle und das Übertragungsmedium über dem unser System mit externen Komponenten interagiert. Dabei wird erklärt, wie und über welche technischen Kanäle unsere fachlichen Ein- und Ausgaben ablaufen. Zur Veranschaulichung wird dies im folgenden UML Deployment-Diagramm dargestellt:

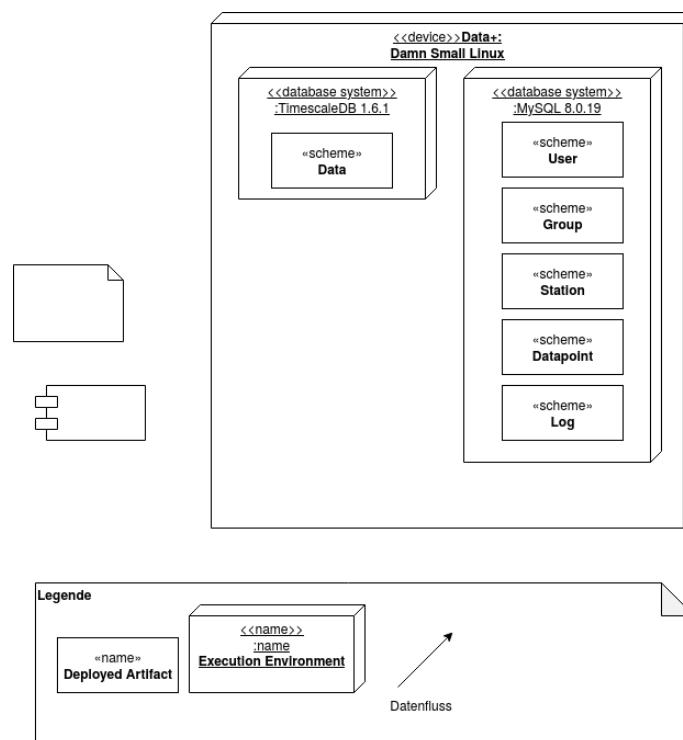


Abbildung 3.2: Technische Kontextabgrenzung

4 Qualitaetsszenarien

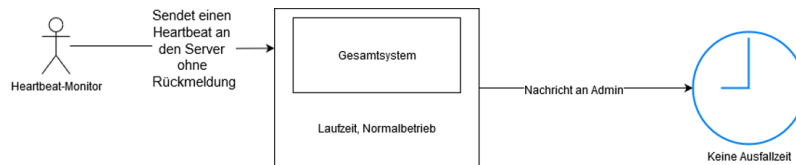


Abbildung 4.1: Verfuegbarkeits-Szenario

Verfuegbarkeits-Szenario Source of Stimulus: Heartbeat Monitor Stimulus: Sendet einen Heartbeat an den Server ohne Rückmeldung Environment: Laufzeit, Normalbetrieb Component: Gesamtsystem Response: Nachricht an Admin Response Measure: ...

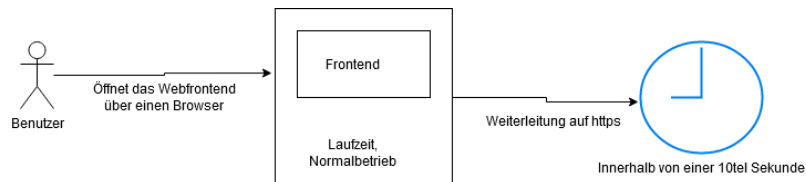


Abbildung 4.2: Effizienz-Szenario

Sicherheits-Szenario Source of Stimulus: Benutzer Stimulus: Öffnet das Webfrontend über einen Browser über http Environment: Laufzeit, Normalbetrieb Component: Frontend Response: Weiterleitung auf https Response Measure: Innerhalb von einer 10tel Sekunde

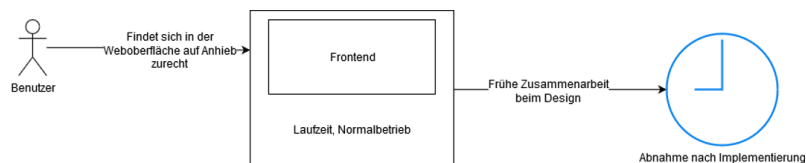


Abbildung 4.3: Nutzbarkeit-Szenario

Nutzbarkeits-Szenario Source of Stimulus: Benutzer Stimulus: Findet sich in der Weboberfläche auf Anhieb zurecht Environment: Laufzeit, Normalbetrieb Component: Frontend Response: Frühe Zusammenarbeit beim Design Response Measure: Abnahme nach Implementierung

Erweiterbarkeits-Szenario Source of Stimulus: Benutzer Stimulus: Neue Funktionen werden zeitnah eingebaut Environment: Laufzeit, Normalbetrieb Component: Frontend, Backend Response: Implementierung ohne negativen Sideeffects Response Measure: Implementierung innerhalb einer Woche

Portierbarkeits-Szenario Source of Stimulus: IT-Abteilung Stimulus: Ein Hardwarewechsel auf eine andere Plattform Environment: Laufzeit, Normalbetrieb Component: Frontend, Backend Response: Implementierung in hardwarenaher Sprache mit Compilern, die Programme für alle Systeme kompilieren Response Measure: Compileroutput muss auf ARM-Cortex 7, Intel x86/x64 unter Linux und Windows funktionieren

4 Qualitaetsszenarien

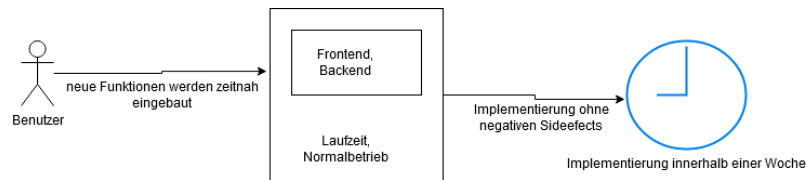


Abbildung 4.4: Effizienz-Szenario

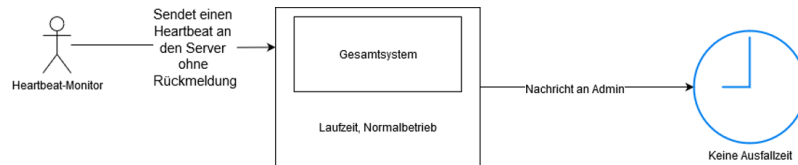


Abbildung 4.5: Effizienz-Szenario

Effizienz-Szenario Source of Stimulus: IT-Abteilung Stimulus: Mehrere Daten sammeln, Hardware soll mind. 500 Datenpunkten aus verschiedenen Stellen sammeln können Environment: Laufzeit, Normalbetrieb Component: Hardware, Software, Backend Response: Einführung von Hardware, bzw. gute Software um Datenpunkte zu sammeln Response Measure:

Hardware-Szenario Source of Stimulus: IT-Abteilung Stimulus: Passende Hardware für die Sprachumgebung Environment: Laufzeit, Normalbetrieb Component: Frontend, Backend Response: Für Frontend und Backend muss eine passende Hardware vorhanden sein Response Measure: Minimalanforderungen für Frontend: Lauffähig auf Raspberry Pi 1; Für Backend: mindestens 512 Mbyte Hauptspeicher, 4 Gbyte Sekundärspeicher und einem Prozessor mit 2 logischen Kernen

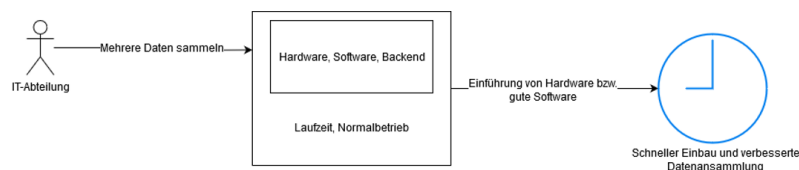


Abbildung 4.6: Effizienz-Szenario

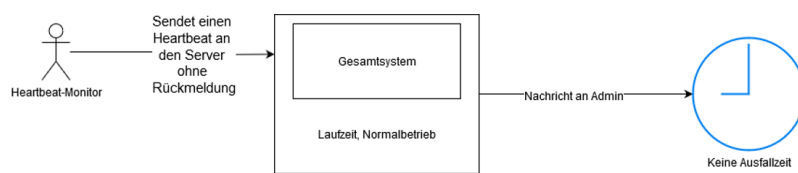


Abbildung 4.7: Effizienz-Szenario

5 Konzept

5.1 Lösungsstrategie

5.2 Bausteinsicht

5.3 Laufzeitsicht

5.4 Verteilungssicht

6 Fazit

Literatur

- [1] Donald E. Knuth. „Computer Programming as an Art“. In: *Communications of the ACM* 17.12 (1974), S. 667–673.

Abbildungsverzeichnis

3.1	Fachliche Kontextabgrenzung	5
4.1	Verfuegbarkeits-Szenario	7
4.2	Effizienz-Szenario	7
4.3	Nutzbarkeit-Szenario	7
4.4	Effizienz-Szenario	8
4.5	Effizienz-Szenario	8
4.6	Effizienz-Szenario	8
4.7	Effizienz-Szenario	9

Tabellenverzeichnis

3.1	Fachliche Kontextabgrenzung der Kommunikationspartner	6
-----	---	---

Listings

Abkürzungsverzeichnis

Anhang

A Erster Abschnitt des Anhangs

In den Anhang gehören „Hintergrundinformationen“, also weiterführende Information, ausführliche Listings, Graphen, Diagramme oder Tabellen, die den Haupttext mit detaillierten Informationen ergänzen.

Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln. Dies hier ist ein Blindtext zum Testen von Textausgaben. Wer diesen Text liest, ist selbst schuld. Der Text gibt lediglich den Grauwert der Schrift an. Ist das wirklich so? Ist es gleichgültig, ob ich schreibe: „Dies ist ein Blindtext“ oder „Huardest gefburn“? Kjift – mitnichten! Ein Blindtext bietet mir wichtige Informationen. An ihm messe ich die Lesbarkeit einer Schrift, ihre Anmutung, wie harmonisch die Figuren zueinander stehen und prüfe, wie breit oder schmal sie läuft. Ein Blindtext sollte möglichst viele verschiedene Buchstaben enthalten und in der Originalsprache gesetzt sein. Er muss keinen Sinn ergeben, sollte aber lesbar sein. Fremdsprachige Texte wie „Lorem ipsum“ dienen nicht dem eigentlichen Zweck, da sie eine falsche Anmutung vermitteln.