

BEUTH HOCHSCHULE
FÜR TECHNIK
BERLIN

University of Applied Sciences

Beuth Hochschule für Technik Berlin Fachgebiet Medieninformatik

Masterarbeit

Cloud-basierte Datenerfassung und Visualisierung für Fahrzeugdaten.

Vorgelegt von:

Jawhar Ben Hadj M'Barek

Franz Mehring Platz. 3

10243 Berlin

Mat.-Nr. 837072

s63338@beuth-hochschule.de

Erstprüfer:

Prof. Dr. Sven Graupner

Zweitprüfer:

Prof. Dr. Siu

Abgabe:

26. November 2018

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Berlin, den

.....

Unterschrift

I. Zusammenfassung/Abstract

i. Zusammenfassung

Der PKW ist das am häufigsten benutzte Fahrzeug in Unternehmen wie beispielsweise Kurier-, Taxi- oder Pflegedienste. PKW Diebstähle und die Kosten für Benzin und Instandhaltung von PKW's sind so immens gestiegen, dass viele Unternehmer genau dadurch der PKW Überwachung wesentlich mehr Aufmerksamkeit und Interesse schenken als ohne.

So viele PKW's überwachen wie man möchte, ohne dass man dabei ist, ganz bequem vom Büro aus. Nicht nur gegen Diebstahl schützen, sondern ebenso Alarm auslösen bei Beschädigungen oder offenen Türen. Außerdem ist auch interessant eine Kostenkontrolle und -analyse über die hohen Benzinkosten zu erhalten. Weiterhin ist es wichtig, dass noch eine genaue Aufstellung der einzelnen Fahrer erfolgt. Welche Route der Fahrer genommen hat sowie die Start- und Stoppzeiten werden detailliert aufgelistet.

ii. Abstract

Automotive knowledge combined with IT Industry. the automotive market today is more driven by the IT industry by new use cases that are coming out from the IT world. People want to do shopping from inside their cars. they want to use services that are connected to the IT world. The challenge for the automotive manufacturers is to compete with a market that's driven by smart devices, which develop in very fast way.

Monitoring the water temperature, the motor rotational speed, the oil level and its temperature and other important information to make sure you can identify any errors or mistakes that may happen on the way, and resolve them on time to save your company from potential business loss.

Act quick and be in control

Once in a while unexpected situations occur, in which case it's important to respond quickly and adjust the plan. Live map and GPS tracking will be extremely useful when finding the nearest vehicle to help other driver nearby and assisting with directions if any driver is off the road.

Inhaltsverzeichnis

I	Zusammenfassung/Abstract	II
i	Zusammenfassung	II
ii	Abstract	II
II	Nomenklatur	VI
i	Formelzeichen	VI
ii	Indizes	VII
1	Einleitung	1
1.1	Zielsetzung	1
2	Technische Grundlagen	2
2.1	Messdaten Extraktion	2
2.1.1	CAN BUS und PEAK	2
2.1.2	OBD und UDS Schnittstellen	2
2.2	Übertragung in Cloud	2
2.2.1	IOT und MQTT	2
2.2.2	AWS IOT und SDK	2
2.2.3	AWS Lambda Funktionen	2
2.2.4	AWS API Gateway	2
2.3	Datenablage und Visualisierung	3
2.3.1	Elasticsearch	3
2.3.2	DynamoDB	3
2.3.3	Kibana	3
2.3.4	JavaFx	3
2.4	Sprachsteuerung	4
2.4.1	AWS Alexa	4
2.4.2	AWS SNS	4
3	Lösungsansätze	5
3.1	Elasticsearch vs. DynamoDB	5
3.1.1	Subsection	5
3.2	Kibana und JavaFx	5

4	Analyse	9
4.1	Anforderungen	9
4.1.1	Echtzeitfähigkeit	9
4.2	dynamische Erstellung von Diagramme	9
5	Design	13
5.1	Use Cases	13
5.1.1	Subsection	13
5.2	Modulendiagramm	14
5.3	Spezifikation von Datenstrukturen	14
5.4	Spezifikation von Schnittstellen	14
5.5	Klassendiagramm	14
6	Implementierung	15
6.1	Architektur	15
6.1.1	Subsection	15
6.2	Realisierungsspezifische Probleme	16
6.3	Testkonzept und Testergebnisse	16
6.4	interessante Systemteile und Kode	16
7	Anwendung	17
7.1	System Bedienung	17
7.1.1	Subsection	17
7.2	gebliebene Fehlermeldungen und Ursachen	18
7.3	Nutzer Reaktion	18
8	Fazit	19
III	Literatur	20
IV	Abbildungsverzeichnis	21
V	Tabellenverzeichnis	22
VI	Anhang	23
i	Section	23
i.1	Subsection	23

II. Nomenklatur

i. Formelzeichen

Formelzeichen	Beschreibung
A	Ausgabeschicht/Ruhelage
c	Koeffizienten im Polynom/Koeffizienten in Teststrecke
\vec{c}	Center-Vektor
\vec{d}	Abstand zwischen zwei Vektoren
E	Eingabeschicht/Fehler
e	Quadratischer Fehler
f	Funktion/Frequenz
$flops$	Fließkommaoperationen
g	Funktion/Parameter
H	Verborgene Schicht bzw. Layer
h	Funktion
I	Netzeingabe
$ I $	Anzahl Beispiele
\vec{i}	Input eines einzelnen Neurons
J	Jacobimatrix
k	Anzahl
m	Anzahl der Ausgänge/ganzzahliges Vielfaches
N	Menge aller Neuronen
n	Anzahl Neuronen/Anzahl Trainingsbeispiele
O	Netzausgabe
o	Output eines einzelnen Neurons
P	Performance
p	Anzahl der Beispiele die erneuert werden
q	Grad des Polynoms/Anzahl der Layer
r	Zeitauflösung
s	Grenze bzw. Schranke
T	Ausgabemuster/Teachinginput
t	Zeit bzw. Zeitpunkt
u	Stellgröße
V	Menge aller Verbindungen

v	Gewicht/Polynom
W	Matrix aller Gewichte
\vec{w}	Eingangsgewichte eines einzelnen Neurons
w	Gewicht
Y	Messwert
y	Regelgröße
\vec{Z}	Innere Zustände eines Neurons
α	Parameter
β	Parameter
Δ	Änderung
δ	Gewichtsänderung
η	Lernrate
Θ	Parameter
μ	Regularisierungsfaktor
ρ	Korrelationswert
τ	Integrationsvariable
∇	Nablaoperator

ii. Indizes

Index	Beschreibung
A	Aktivierung
alt	alt/aus vergangenem Zeitschritt
i	Zählindex
j	Zählindex
k	Zählindex
$last$	Vergangene Werte
$Modell/M$	Größen die aus dem Modell berechnet wurden
max	Maximal
o	Ausgabe
P	Propagierung
p	Prediction
p	Ein bestimmtes Trainingsmuster
q	Zählindex
$rand$	Zufällig
s	Statisch
T	Teachinginput
$Train$	Training
Val	Validation

1. Einleitung

Einleitung Einleitung Einleitung Einleitung Einleitung Einleitung Einleitung Einleitung

1.1. Zielsetzung

Einleitung Einleitung Einleitung Einleitung Einleitung Einleitung Einleitung Einleitung

2. Technische Grundlagen

[illegible]

2.1. Messdaten Extraktion

Eine Section

2.1.1. CAN BUS und PEAK

2.1.2. OBD und UDS Schnittstellen

2.2. Übertragung in Cloud

2.2.1. IOT und MQTT

Eine Subsection

2.2.2. AWS IOT und SDK

Eine Subsection

2.2.3. AWS Lambda Funktionen

Eine Subsection

2.2.4. AWS API Gateway

Eine Subsection und noch eine neue Seite

2.3. Datenablage und Visualisierung

2.3.1. Elasticsearch

2.3.2. DynamoDB

Eine Subsection

2.3.3. Kibana

Eine Subsection

2.3.4. JavaFx

Eine Subsection

2.4. Sprachsteuerung

2.4.1. AWS Alexa

Eine Subsection

2.4.2. AWS SNS

Eine Subsection

3. Lösungsansätze

Hier ein anderes Kapitel Viele Zitate: [6] [1] [5] [3] [2] [4]

3.1. Elasticsearch vs. DynamoDB

The most important thing about Elasticsearch is that Elasticsearch supports full text search and that makes it different from the normal databases, which are very much black and white, we just store stuff and retrieve it. In Full text search we look more for the concept, not a specific word in singular or in plural but we are looking for the concept or the meaning, which is very complicated to achieve with the normal databases for many reasons: building complicated queries, much more computing time by searching, long time for getting the response and by returning the result and not very performant like with Elasticsearch.

Common Databases are focusing on how to store the data comparing to Elasticsearch that allows us to make sense of our data, explore them and be able to ask questions in a simple way without writing complicated Query. Elasticsearch give us the ability to make data exploration, to go ahead and ask questions of the saved datas and get results in milliseconds and be ready to visulise and make analytics.

3.1.1. Subsection

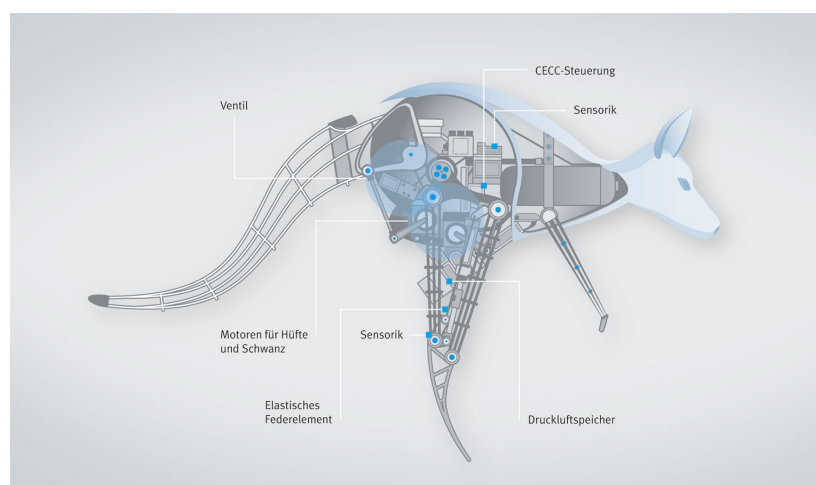


Abbildung 3.1.: Ein Kangaroo

Spalte1	Spalte2	Spalte3
1	2	3

Tabelle 3.1.: Eine Tabelle

3.2. Kibana und JavaFx

Eine Subsection

Eine neue Seite

Noch eine neue Seite

und noch eine neue Seite

4. Analyse

Hier ein neues Kapitel Viele Zitate: [6] [1] [5] [3] [2] [4]

4.1. Anforderungen

Eine Section

4.1.1. Echtzeitfähigkeit

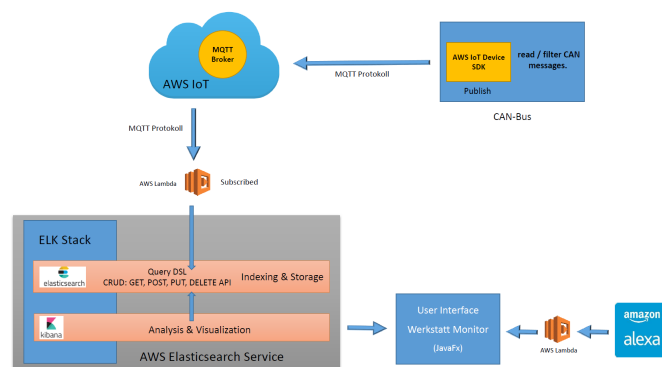


Abbildung 4.1.: das Architekturbild

Spalte1	Spalte2	Spalte3
1	2	3

Tabelle 4.1.: Variation über Zeit

Hier eine neue Tabelle

Spalte1	Spalte2	Spalte3
1	2	3

Tabelle 4.2.: new added Tabelle

4.2. dynamische Erstellung von Diagramme

Eine Subsection

Eine neue Seite

Noch eine neue Seite

und noch eine neue Seite

Hier ein neues Kapitel Viele Zitate: [6] [1] [5] [3] [2] [4]

5.1. Use Cases

Eine Section

5.1.1. Subsection

Hier ist ein Bild:

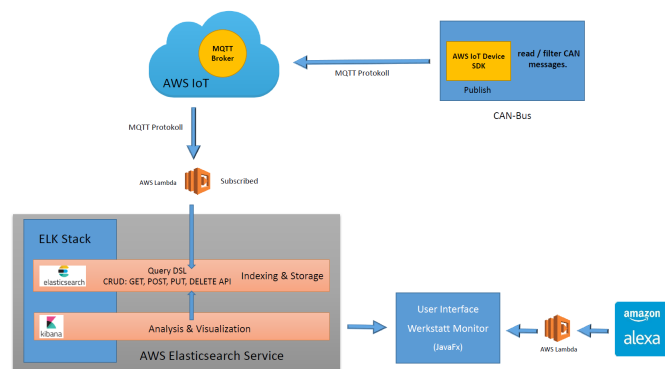


Abbildung 5.1.: Affe Bild

[illegible]

Spalte1	Spalte2	Spalte3
1	2	3

Tabelle 5.1.: Quadratewurzel Skalierung

[illegible]

Tabelle 5.2.: Logarithmische Skalierung

Eine Subsection

Eine Subsection

Eine Subsection

Eine Subsection

Hier ein neues Kapitel Viele Zitate: [6] [1] [5] [3] [2] [4]

6.1. Architektur

Eine Section

6.1.1. Subsection

Hier ist ein Bild:

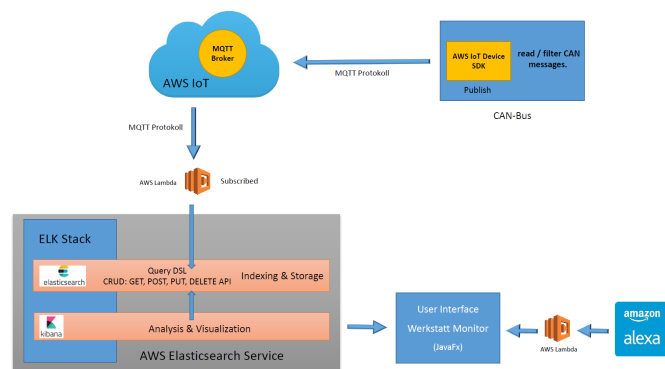


Abbildung 6.1.: Affe Bild

[illegible]

Spalte1	Spalte2	Spalte3
1	2	3

Tabelle 6.1.: Quadratewurzel Skalierung

[illegible]

Tabelle 6.2.: Logarithmische Skalierung

Eine Subsection

Eine Subsection

Eine Subsection

Hier ein neues Kapitel Viele Zitate: [6] [1] [5] [3] [2] [4]

7.1. System Bedienung

Eine Section

7.1.1. Subsection

Hier ist ein Bild:

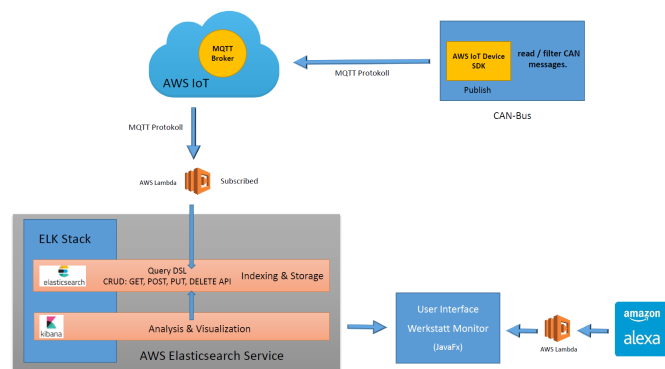


Abbildung 7.1.: Affe Bild

[illegible]

Spalte1	Spalte2	Spalte3
1	2	3

Tabelle 7.1.: Quadratewurzel Skalierung

[illegible]

Tabelle 7.2.: Logarithmische Skalierung

Eine Subsection

Eine Subsection

8. Fazit

Hier schreiben wie gut alles war.

III. Literatur

- [1] Alex Krizhevsky, Ilya Sutskever und Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, S. 1097–1105.
- [2] Steve Lawrence u. a. “Face recognition: A convolutional neural-network approach”. In: *IEEE transactions on neural networks* 8.1 (1997), S. 98–113.
- [3] Warren S. McCulloch und Walter Pitts. “A logical calculus of the ideas immanent in nervous activity”. In: *The bulletin of mathematical biophysics*. Bd. 5. 1943, S. 115–133.
- [4] Stefan Miesbach. “Bahnführung von Robotern mit neuronalen Netzen”. Diss. Technical University Munich, Germany, 1995.
- [5] *Neural Network Time-Series Prediction and Modeling (Matlab Userguide)*. <https://de.mathworks.com/help/nnet/gs/neural-network-time-series-prediction-and-modeling.html>. Zugriff: 05.07.2017.
- [6] Dan Patterson. *Künstliche neuronale Netze*. zweite Auflage. München: Prentice Hall, 1997.

IV. Abbildungsverzeichnis

3.1	Ein Kangoroo	5
4.1	das Architekturbild	9
5.1	Affe Bild	13
6.1	Affe Bild	15
7.1	Affe Bild	17

V. Tabellenverzeichnis

3.1	Eine Tabelle	5
4.1	Variation über Zeit	9
4.2	new added Tabelle	9
5.1	Quadratwurzel Skalierung	13
5.2	Logarithmische Skalierung	14
6.1	Quadratwurzel Skalierung	15
6.2	Logarithmische Skalierung	16
7.1	Quadratwurzel Skalierung	17
7.2	Logarithmische Skalierung	18

VI. Anhang

Das hier ist der Anhang

i. Section

i.1. Subsection