

Pflichtenheft PSE - Automotive Remote Sensor Data Management (aRSDM)

Vladimir Bykovski Jonas Haas David Grajzel
Nicolas Schreiber Valentin Springsklee Yimeng Zhu

20. November 2015

Inhaltsverzeichnis

1	Einleitung	1
1.1	Aufgabenstellung	1
2	Zielbestimmung	2
2.1	Musskriterien	2
2.1.1	/MK1010/Protokollierung	2
2.1.2	/MK1020/Erweiterbarkeit	2
2.1.3	/MK1030/RemoteAcces	2
2.1.4	/MK1040/Anzeigeelemente	2
2.1.5	/MK1050/Statistiken	2
2.1.6	/MK1060/Fahrinformation	2
2.1.7	/MK1070/Durschnittsverbrauch	3
2.1.8	/MK1080/Restkilometer	3
2.1.9	/MK1090/Einparkhile	3
2.1.10	/MK1100/POI	3
2.2	Wunschkriterien	3
2.2.1	/MK2010/Smartphone Utilization	3
2.2.2	/MK2020/Personalisierung	3
2.3	Abgrenzungskriterien	3
2.3.1	/MK3010/Erweiterung	3
2.3.2	/MK3020/Unidirektional	3
2.3.3	/MK3030/Bildauswertung	3
2.3.4	/MK3040/Android	3
2.3.5	/MK3050/Flotte	4
3	Produkteinsatz	5
3.1	Anwendungsbereiche	5
3.2	Zielgruppen	5
3.3	Betriebsbedingungen	5
4	Produktumgebung	6
4.1	Software	6
4.2	Hardware	7
5	Produktfunktionen	8
5.0.1	/FA-1010/Dateninput	8
5.0.2	/FA-1020/Vergangene-Daten	8
5.0.3	/FA-1030/Karte	8

5.0.4	/FA-1040/Zustandswarnungen	8
5.0.5	/FA-1050/Einparkhilfe	8
5.1	Autos	9
6	Produktdaten	10
7	Produktleistungen	11
8	Benutzerschnittstelle	12
8.1	GUI	12
8.2	Schnittstelle für Netzwerk	12
9	Globale Testfälle	13
9.1	Netzwerk	13
9.2	Pi im Auto	13
9.3	Endgeräte	13
10	Qualitätsbestimmung	14

1 Einleitung

In der modernen Informationsgesellschaft fallen immer mehr nutzbare Daten an. Seien es Daten von sozialen Netzwerken und Daten von Personen über Nutzung und Gewohnheiten allgemein, oder Daten technischer Art, wie Sensordaten in Anlagen oder Automobilen. Mehr und mehr Sensorik findet Einsatz. Oft ist es sehr interessant diese Daten auszuwerten und visuell zu repräsentieren. Tablets und Smartphones sind omnipräsent. Jeder hat dauerhaft Zugriff auf einen leistungsfähigen Rechner mit hochauflösendem Display. Diese Geräte eignen sich also prima zur Anzeige von Daten. Unabhängig von der Art der Daten, müssen sie gesammelt, evtl. vorverarbeitet und bereitgestellt werden. Dazu bedarf es einer generischen Schnittstelle. In dem Rahmen von aRSDM sollen Sensordaten gesammelt und verteilt werden. Genauer gesagt, die über den OBD2 Anschluss verfügbare Daten in einem PKW. Es ist eine (abstrakte) Komponente zwischen Sensoren und Endgerät gefragt, die sich für beliebige, nicht unbedingt schon bekannte Anwendungsszenarien erweitern lässt. Ein Rahmenwerk, das Daten sammelt, speichert und verteilt.

1.1 Aufgabenstellung

Smartphones und Tablets des Users sind die sogenannten Endgeräte. Auf ihnen erfolgt eine Grafische Darstellung von Sensordaten. Die Quelle der Daten ist entfernt und die Daten werden gelangen über Netzwerk auf das Smartphone. RSDM bietet eine API, die von Dritten für weitere Szenarien genutzt werden kann. Als Anwendung liefern wir ein automotive Modul, welches Sensordaten, die über den CAN Bus eines PKW verfügbar sind, verarbeitet.

2 Zielbestimmung

Die Software aRSDM soll Sensordaten von einem Auto sammeln, die über den CAN Bus auszulesen sind speichern und Endgeräten zur Darstellung zur Verfügung stellen. Dabei soll eine API implementiert und genutzt werden, die auch auf weitere Anwendungen erweiterbar ist.

Es soll die Darstellung im Browser Google Chrome™ auf Endgeräten erfolgen.

2.1 Musskriterien

Allgemeine Funktionen

2.1.1 /MK1010/Protokollierung

Der Nutzer soll Daten aus der Vergangenheit zu jeder Zeit abrufen können.

2.1.2 /MK1020/Erweiterbarkeit

Drittentwickler können Module für die Verarbeitung und Auswertung weiterer Sensordaten schreiben.

2.1.3 /MK1030/RemoteAccess

Falls Netzwerkverbindung besteht, können entfernte Nutzer die Sensordaten einsehen.

2.1.4 /MK1040/Anzeigeelemente

Der Nutzer kann Anzeigeelemente ein- und ausblenden.

2.1.5 /MK1050/Statistiken

Der Benutzer kann sich Statistiken der gespeicherten Sensorwerte anzeigen lassen.

Automotive Kontext

2.1.6 /MK1060/Fahrinformation

Der Benutzer bekommt Fahrinformation (Glossar) angezeigt.

2.1.7 /MK1070/Durschnittsverbrauch

Die Software zeigt den Durchschnittsverbrauch des Fahrzeugs an.

2.1.8 /MK1080/Restkilometer

Die Software zeigt die verfügbaren Restkilometer bis zur nächsten Tankfüllung an. Diese werden aufgrund des Fahrverhaltens berechnet.

2.1.9 /MK1090/Einparkhile

Dem Fahrer steht eine Einparkhilfe mit Augmented Reality Features zur Verfügung.

2.1.10 /MK1100/POI

Bei bestimmten Sensorwerten (Tank/Motor) kann der sich Nutzer die nächsten Tankstellen / Werstätten einblenden.

2.2 Wunschkriterien

2.2.1 /MK2010/Smartphone Utilization

Die Software verwendet auch Sensordaten aus dem Endgerät.

2.2.2 /MK2020/Personalisierung

Aggregierte Funktionen sind personalisiert verfügbar.

2.3 Abgrenzungskriterien

2.3.1 /MK3010/Erweiterung

Für andere Sensorquellen werden keine Treiber bereitgestellt.

2.3.2 /MK3020/Unidirektional

Die Software liest und interpretiert Daten, sendet keine Daten an Quellen zurück.

2.3.3 /MK3030/Bildauswertung

Karabilder werden von der Software nicht interpretiert.

2.3.4 /MK3040/Android

Die Visualisierung der Daten erfolgt nur im Browser, wir stellen keine gesonderte App zur Verfügung.

2.3.5 /MK3050/Flotte

Der Zugriff auf verschiedene Fahrzeuge wird verwaltet.

3 Produkteinsatz

Das Produkt dient zur Speicherung und Distribution von Daten mehrerer verteilter Sensoren eines Autos. Dies beinhaltet z.B. Informationen wie die Geschwindigkeit, Motortemperatur, Raddrehzahl oder ähnliches. Damit bietet es die Möglichkeit diese Daten z.B. auf einem Handydisplay oder einem Dashboard anzuzeigen

3.1 Anwendungsbereiche

- Dieses Produkt dient zum Speichern und Anzeigen von Sensordaten, die im Auto verfügbar sind.

3.2 Zielgruppen

- Private Autofahrer mit oder ohne tiefere technische Kenntnisse. Firmen mit Automobilflotten.

3.3 Betriebsbedingungen

- Auto muss über OBD2 Anschluss verfügen.
- Es wird über den Autostrom laufen.
- Für Remote Zugriff muss Netzwerkverbindung vorhanden sein.

4 Produktumgebung

Das Rahmenwerk und die Module werden in TypeScript programmiert. Der Server, in node.js.

- Eine Client-Server Architektur mit dem Thin Client Konzept
- Auf dem Server läuft der Teil der Software, der die Sensordaten empfängt, einpflegt und über WebRTC an die Endgeräte versendet.
- Auf der Userseite laufen einige verschiedene Applikationen neben Unserer. Diese sollen sich jedoch gegenseitig nicht beeinflussen.

Testumgebung

- Testfahrzeug: Opel Astra H 1.6 Bj 2005
- Modellauto Nicolas
Ein Modellauto der Firma ITK-Engineering AG, angetrieben über einen Elektromotor. Jegliche Daten, unter anderem von Ultraschallsensoren, liegen über CAN an

4.1 Software

- Serverseite
Ein Webserver mit einer Datenbank
Aufgebaut auf einer Linux-Distribution
- Clientseite
Web Browser:
 - Google Chrome Version 46
 - Safari Version 9
 - Firefox Version 42
 - Google Chrome Android Version 14.10Getestet mit:
 - Mac OS X
 - Fedora 22
 - Windows 7

4.2 Hardware

- Serverseite
Kleincomputer mit:
 - Broadcom BCM2836 Arm7 Quad Core Processor (900MHz)
 - 1GB RAM
 - 4 x USB 2 portsCan-Modul
- Clientseite
Standardrechner (min. 1 GHz und 2 GB Ram)
Android Phone (min. 1 GHz und 1 GB Ram)

5 Produktfunktionen

5.0.1 /FA-1010/Dateninput

Wenn Daten über Bluetooth ankommen pflegt die Software diese Daten in eine Datenbank ein.

5.0.2 /FA-1020/Vergangene-Daten

Wenn der Nutzer es wünscht zeigt die Software ihm die gespeicherten Daten an.

5.0.3 /FA-1030/Karte

Der Benutzer kann sich vom System einen aktuellen Kartenausschnitt der Umgebung anzeigen lassen.

/FA-1030/Werkstatt/Tankstellen

Die Software kann auf dieser Karte auch Werkstätte, Tankstellen und Restaurants anzeigen.

/FA-1030/Niedriger Tankfüllstand

Bei niedrigem Tankfüllstand werden beim öffnen dieser Karte automatisch die Tankstellen in der Nähe angezeigt.

/FA-1030/Weitere Probleme

Bei weiteren Problemen wie einer Kontrollleuchte werden beim öffnen dieser Karte die Werkstätten in der Nähe angezeigt.

5.0.4 /FA-1040/Zustandswarnungen

Das System überwacht und reagiert auf kritische Sensorwerte und Warnungen.

5.0.5 /FA-1050/Einparkhilfe

Beim rückwärts Einparken aktiviert die Software eine Rückfahrkamera und indiziert die aktuelle Fahrbahn.

/FA-1050/Ultraschallsensoren

Außerdem werden die Ultraschallsensordaten angezeigt.

5.1 Autos

6 Produktdaten

7 Produktleistungen

8 Benutzerschnittstelle

8.1 GUI

8.2 Schnittstelle für Netzwerk

9 Globale Testfälle

9.1 Netzwerk

9.2 Pi im Auto

9.3 Endgeräte

10 Qualitätsbestimmung