**Arc42**

**1.** **Introduction and Goals**

Dieses Dokument beschreibt das System Schadstoffklassenrechner und alle damit verbundene Schnittstellen zu externen Systemen und Librarys.

Das System nimmt eine Anfrage von der Printabteilung entgegen, übernimmt dann den Ablauf in dem es Daten von der Logistikabteilung und Fahrzeugentwicklung entgegen nimmt und dies an eine Simulationssoftware gibt. Am Ende werden die fertig berechneten Simulationsdaten an den Print zurückgegeben.

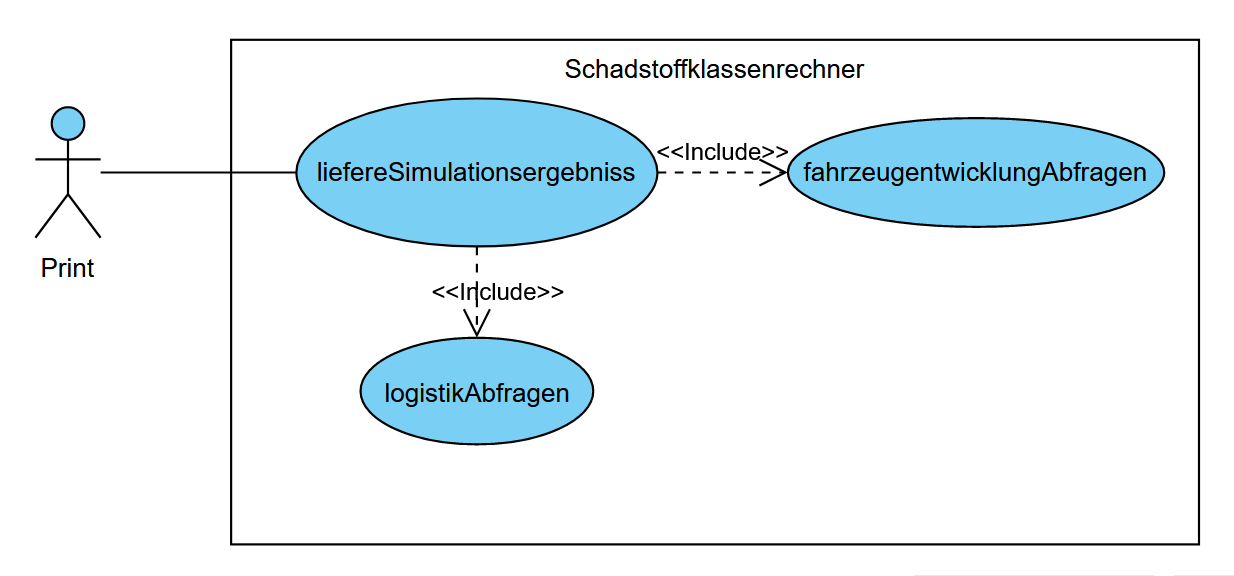
Die Ziele für das System sind unter anderem:

- Das System soll für jede Anfrage die richtige Antwort geben, da es eine rechtliche Vorgabe ist muss die Antwort zwingend richtig sein.

- Das System soll einfach wartbar und erweiterbar sein, da es vorkommen wird dass sich die Gesetzeslage ändert, wodurch auch das System angepasst werden muss.

- Es muss sichergestellt werden, dass keine kritische Daten nach außen treten, wegen des Risikos der Wirtschaftsspionage.

**1.1. Requirements Overview**



F1: Das System bietet nach außen genau einen Zugangspunkt, der Print kann eine Anfrage starten und alles weitere wird dann intern im System verarbeitet oder direkt vom System an andere Systeme geleitet, ohne dass ein User eine Aktion durchführen muss.

**1.2. Qualitätsziele**

|  |  |
| --- | --- |
| Qualität | Beschreibung |
| Rechtsicherheit | Da ein EU-Gesetz vorliegt muss das System immer richtige Ergebnisse liefern. |
| Erweiterbarkeit | Das System muss eine hohe Erweiterbarkeit besitzen, da es wahrscheinlich ist, dass die gesetzlichen Anforderungen ändern. |
| Performance | Das System soll schnelle Ergebnisse liefern, da erst wenn die Ergebnisse beim Print sind, die LKWs das Geländer verlassen dürfen. |
| Security | Es sollen keine Informationen über den Aufbau der Fahrzeuge nach außen treten, um Industriespionage zu verhindern. |

**1.3. Stakeholder**

Die Stakeholder sind der Leiter der IT-Abteilung, der Projektleiter und die Leiter der Print-, Logistik- und Fahrzeugentwicklungsabteilung

**2. Randbedingungen**

**2.1 Technische Randbedingungen**

Externe APIs:  
-**Dokumentendruck**: stellt die Anfrage an unser System, erhält das Ergebnis und druckt dieses  
-**Fahrzeugdaten-API**: nimm Konfiguration an und gibt Messdaten zurück  
-**Logistik-API**: nimmt Code, Konfiguration und Produktionsdatum entgegen und liefert Gewicht und Abmessungen zurück  
-**Vecto-API**: Simuliert CO2-Werte und Umweltrichtlinien Klasse für Nutzfahrzeuge und liefert die Ergebnisse. Hängt von Ergebnissen der anderen APIs ab

**2.2 Organisatorische Rahmenbedingungen**

Das System ist sehr wichtig für den Geschäftsablauf. Entsprechend ist das Projekt sehr kritisch.

**2.3 Konventionen**

Das System hängt von den aktuellen Umweltrichtlinien für Nutzfahrzeuge ab. Änderungen dieser Richtlinien müssen sofort auch in das System implementiert werden.

**3. Kontextabgrenzung**

**3.1 Fachlicher Kontext**

Das System wird vom Dokumentendruckt genutzt. Der Dokumentendruck muss aus rechtlichen Gründen für neue Fahrzeuge Dokumente drucken und benötigt hierfür die Daten unseres Systems. Der Dokumentendruck schickt an unser System eine Konfiguration und erhält die nötigen Simulationsergebnisse

**3.2. Technischer Kontext**

Unser System ist auf technischer Ebene der Vermittler zwischen drei APIs. Wir erhalten Daten vom Dokumentendruckt und schicken einen Teil der Daten an die Fahrzeugdaten-API und die Logistik-API.  
Die Fahrzeugdaten-API erhält einige Daten, wie Konfigurationen und liefert Gewicht und Abmessung zu dem zugehörigen Fahrzeug zurück.  
Die Fahrzeugdaten API liefert für alle Komponenten Messdaten.  
Die zurückgelieferten Ergebnisse schicken wir an Vecto-API, welches uns die Endsimulationsergebnisse liefert. Diese werden nun an den Dokumentendruck zurückschicken.  
Der Hersteller der APIs entwickelt diese ständig weiter und passt sie an neue Gegebenheiten und Umstände an. Zudem ist die API extern und darf somit nicht alle Daten erhalten!

**4. Lösungsstrategie**

Zur Lösung des Problems wurde eine Software entwickelt, welche eine API zur Verfügung stellt, welche die benötigten Daten entgegennimmt und an die externen APIs weiterleitet. Die Simulationsergebnisse werden in einer strukturierten Form zurückgeliefert.

Hier sind die unsere technischen Architekturentscheidungen zu sehen:

<Link zu den MDs>

**5. Bausteinsicht**

Die beiden Diagramme zeigen verschiedene Ebenen der Architektur. In der ersten Ebene werden die Beziehungen zwischen den einzelnen externen Systemen mit unserem System visualisiert.  
In der zweiten Ebene wird die Kommunikation zwischen den Schnittstellen auf einer tieferen Ebene modelliert.

<Diagramme>

6**. Laufzeitsicht**

**9. Entwurfsentscheidungen**

Hier sind alle Architekturentscheidungen dokumentiert:

<Link zu den ADRs>

**10. Qualitätsanforderungen**

1. Datenqualität  
Da es sich um ein für den Geschäftsbetrieb sehr kritisches System handelt, müssen die gelieferten Daten sehr akkurat sein und dürfen keine Fehler enthalten. Die Daten müssen alle gesetzlichen Bestimmungen erfüllen

2. Verfügbarkeit  
Das System muss sehr ausfallsicher sein, da Ausfälle weitere Auswirkungen haben und den Geschäftsbetrieb aufhalten können. Systemfehlern muss also sehr sorgfältig vorgebeugt werden und Wartungen müssen weitgehend vermieden werden.

3. Datenschutz  
Die Daten enthalten Geschäftsgeheimnisse, die nicht nach außen gelangen dürfen. Entsprechend gibt es Anforderungen an die Zugriffssicherheit.

**11. Risiken und technische Schulden**

1. Umweltrichtlinien  
Da das System anhand von rechtlichen Richtlinien arbeiten muss, muss sichergestellt werden, dass diese erfüllt werden. Abmahnungen und Strafzahlungen müssen dringend vermieden werden.

2. Schutz der sensiblen Daten  
Die verarbeiteten Daten sind sensibel und dürfen nicht nach außen gelangen. Da eine der APIs extern verwaltet wird, muss sichergestellt werden, dass nur genau die Daten geliefert werden, die vorgesehen sind. Fehler könnten hohe finanzielle Schäden verursachen