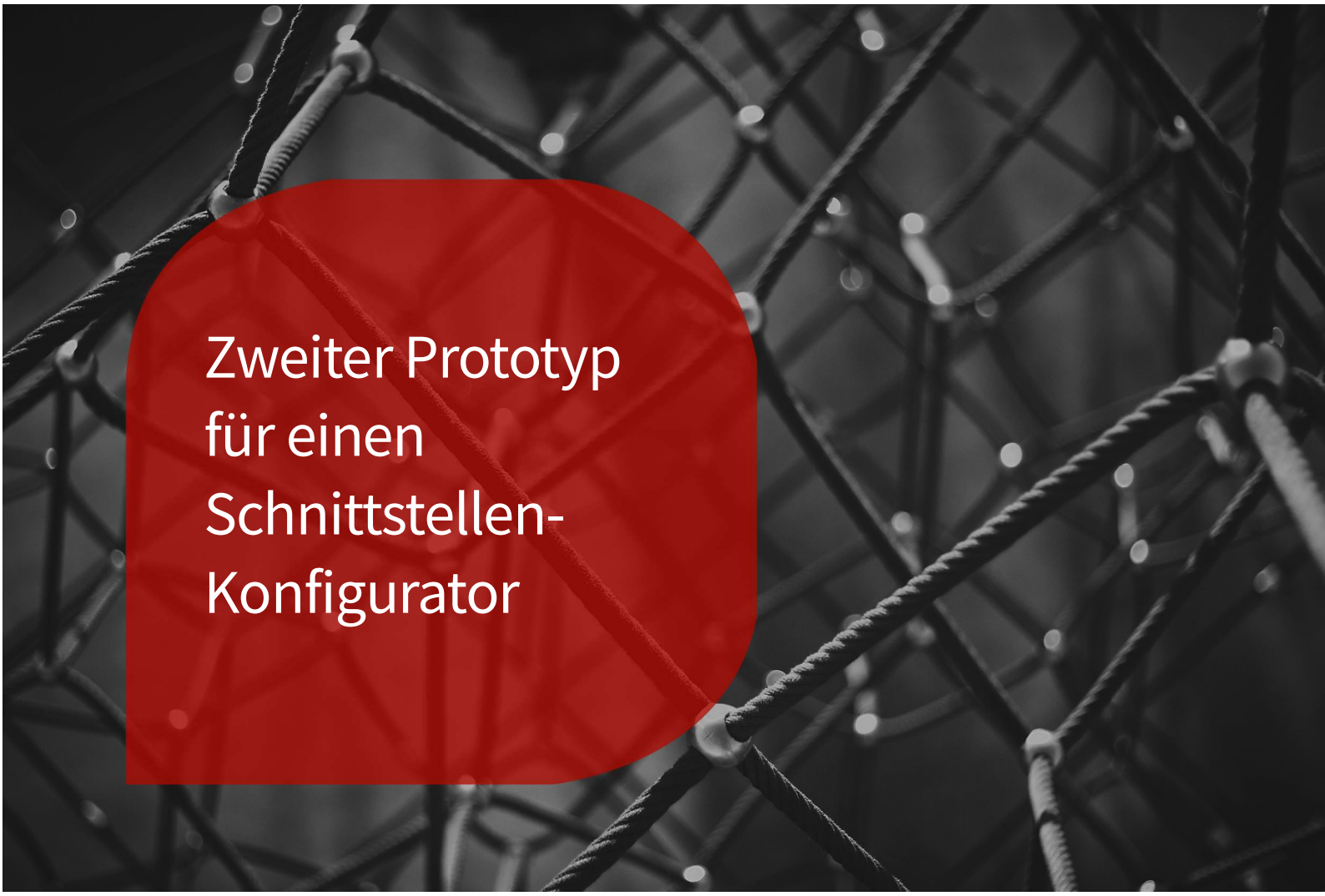
Mittelstand 4.0
Kompetenzzentrum
IT-Wirtschaft



Zweiter Prototyp für einen Schnittstellen-Konfigurator

www.itwirtschaft.de

Mittelstand-Digital

Gefördert durch:
Bundesministerium für Wirtschaft und Klimaschutz
aufgrund eines Beschlusses des Deutschen Bundestages

Inhalt

1 Einleitung 3

Aufbau des Dokuments 4

2 Systembeschreibung 4

3 Hinzugefügte Komponenten 6

3.1 Backend 6

3.2 Backend Webinterface 7

3.3 Exporter 11

4 Aktualisierte Komponenten 13

4.1 Importer 13

4.2 Datenbank 13

4.3 Frontend 15

4.4 Schnittstelle 16

5 Interner Testzugang 17

6 Abbildungsverzeichnis 17

7 Kontakt 19

1 Einleitung

Das grundlegende Ausgangsszenario ist die Kooperation von mittelständischen IT-Unternehmen und deren Softwarelösungen auf Basis von offenen, technischen Standards. Bei der Zusammenführung von Systemen besteht häufig ein Konflikt zwischen proprietären Schnittstellenlösungen beteiligter Softwareunternehmen. Daher erfolgt die Vernetzung auf Grundlage offener, herstellerunabhängiger und lizenzkostenfreier Standards für KMUs der IT-Branche.

Das KIW Schnittstellen Set beruht auf offenen Standards (zur fachlichen Modellierung insb. die UN/CEFACT CCL) und gängigen Formaten. Die Realisierung der API Endpunkte sowie die Nutzung der API kann in jeder geeigneten Entwicklungssprache erfolgen. Auf diese Weise sollen lose gekoppelte und flexibel erweiterbare Integrationsszenarien unterstützt werden, in denen mehrere Business Software Anwendungen jeweils unabhängig voneinander Daten austauschen können. Dies geschieht über generisch definierte KIW-Objekte und KIW-Methoden.

Im Folgenden wird die zweite Version eines Schnittstellen-Konfigurator-Prototypen beschrieben. Er dient der einfachen Generierung von Auszügen des Schnittstellenkatalogs. Der Schnittstellen-Konfigurator wird als Onlineplattform bereitgestellt, so dass Systemabhängigkeit und Standortgebundenheit vermieden werden.

Moderne Softwareunternehmen haben die Wichtigkeit von Softwareschnittstellen in ihren Produkten bereits erkannt und dementsprechend reagiert (vgl. Abbildung 1: Wachstum von Web-APIs). Abhängig vom Use Case Szenario kommen in den Firmen verschiedene APIs und damit auch unterschiedliche Datenaustauschformate [1] zum Einsatz. Zur besseren Wiederverwendbarkeit sind diese verwendeten Formate meist standardisiert.

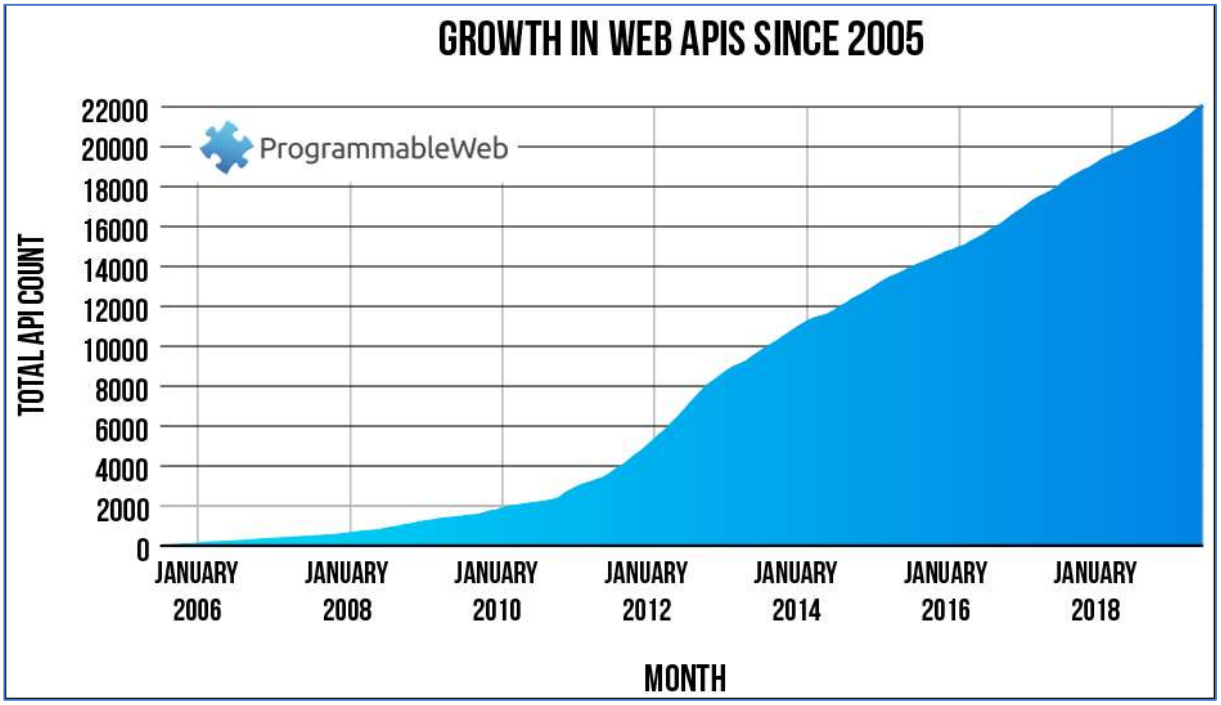


Abbildung 1: Wachstum an Web-APIs aus dem öffentlichen Verzeichnis für Web-APIs von Programmableweb.com

Quelle: https://www.programmableweb.com/news/apis-show-faster-growth-rate-2019-previous-years/research/2019/07/17

Aufbau des Dokuments

Das vorliegende Dokument ist in zwei Kapitel aufgeteilt. Im ersten Kapitel “Systembeschreibung” wird ein Überblick über die Problemstellung und die Lösung die das Tool „KIW-Konfigurator“ dafür bereitstellt vorgestellt.

Im zweiten Kapitel “Hinzugefügte Komponenten” werden Komponenten vorgestellt, die dem Konfigurator seit der letzten Version hinzugefügt wurden.

Im dritten Kapitel “Aktualisierte Komponenten” werden die Änderungen an bestehenden Komponenten seit der letzten Prototypenversion beschrieben.

Im vierten Kapitel “Interner Testzugang” sind Zugangsdaten festgehalten, die verwendet werden können, um den aktuellen Stand des Konfigurators zu testen, und so die Möglichkeit zu haben Feedback für weitere Versionen geben zu können.

2 Systembeschreibung

In diesem Kapitel geben wir einen Überblick über die Problemstellung und die Lösung die das Tool „KIW-Konfigurator“ bereithält.

Der KIW Schnittstellenkatalog bietet eine standardisierte Objektbibliothek die mittelständischen IT-Softwareentwicklern bei Integrationsszenarien unterstützen soll, in denen Business Anwendungen unabhängig voneinander Daten miteinander austauschen.

Das KIW versucht Kooperationen zwischen mehreren IT-Softwareanbietern zu bilden und zu unterstützen, sogenannte Konsortien. Diese benötigen oftmals nur einen Bruchteil der im KIW Schnittstellenkatalog bereitgestellten Objekte. Da das tatsächlich verwendete Subset des Schnittstellenkatalogs nur einen Auszug der Gesamtbibliothek darstellt, muss auch die Dokumentation des Konsortiums nur den tatsächlich verwendeten Ausschnitt umfassen. Die ständige Zusammenstellung der neuen Auszüge aus der Gesamtbibliothek ist jedoch sehr arbeitsaufwendig und kann komplexe Abhängigkeiten beinhalten.

Der KIW Konfigurator verringert den Aufwand der Erstellung von Teilauszügen der Gesamtbibliothek, indem es diesen Prozess automatisiert. Das Tool ermöglicht den Import und den Export von Objektauszügen mit entsprechenden Abhängigkeiten.

Der Konfigurator ist als Webanwendung umgesetzt und damit plattformübergreifend einsetzbar. Die Objekte werden zunächst durch den Import eines Auszuges der Objektbibliothek in generalisierter Form in der Datenbank gespeichert. Danach sind diese durch die Webanwendung dargestellt. Über die Webanwendung lässt sich dann eine entsprechende Auswahl der Objekte und Attribute machen inklusive aller Abhängigkeiten. Durch den Export wird die Auswahl vom Konfigurator in einen der Beschreibungssprache OpenAPI validen Auszug erzeugt.

Abbildung 2: Ablaufprozess des Konfigurators, vom Import bis zum Export.

Für den Import wird ein Node.js Skript verwendet mit welchem die initiale Objektspezifikation automatisiert in die Datenbank importiert wird. Die Daten werden durch einen PHP-Endpunkt dem Webinterface bereitgestellt, in welcher die Konfiguration mittels Benutzerinteraktion stattfindet. Die Auswahl wird im Laufe des Exportprozesses an einen weiteren Endpunkt übersandt, welcher eine OpenAPI valide YAML-Datei erzeugt, die dem ausgewählten Subset der Objektbibliothek entspricht. Der dann erzeugte Auszug wird zum Download zur Verfügung gestellt.

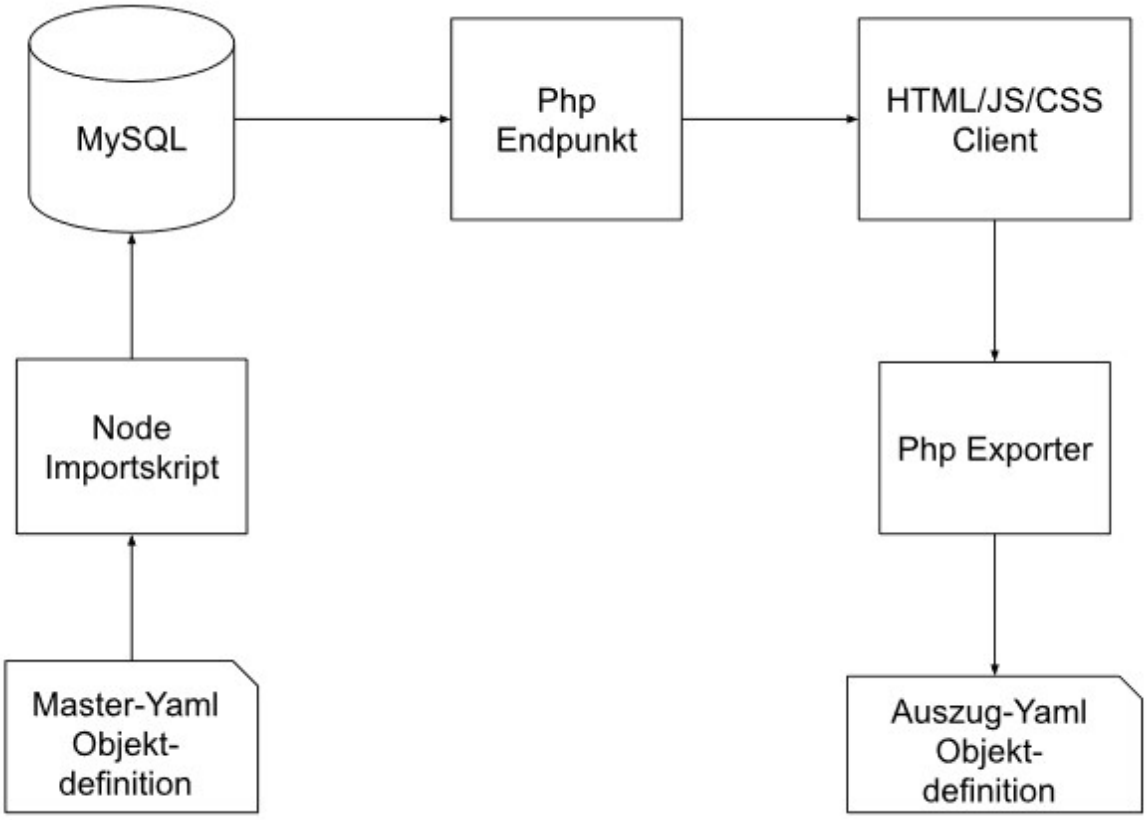


Abbildung 3: Ablaufprozess des Konfigurators aus technischer Sicht, vom Import bis zum Export.

3 Hinzugefügte Komponenten

In diesem Kapitel werden Komponenten vorgestellt, die dem Konfigurator seit der letzten Version hinzugefügt wurden.

3.1 Backend

Das Backend ist eines der neu hinzugefügten Komponenten für den zweiten Prototypen des Schnittstellen Konfigurators. Es ermöglicht die Anpassung des KIW Schnittstellen-Sets in der Datenbank über ein Webinterface. Es ist über den Pfad domain.tld/backend aufrufbar.

Das Backend ermöglicht grundlegend folgende Funktionen:

- Anzeige aller Objekte/Attribute/Standards
- Anlegen von Objekten
- Anlegen von Attributen
- Editieren von Objekten
- Editieren von Attributen
- Löschen von Objekten
- Löschen von Attributen´

Authentifizierung

Der Zugang zum Backend ist nur autorisierten Personen erlaubt. Beim Aufruf der Backendseite wird per Benutzername und Passwort die Zugangsberechtigung überprüft.

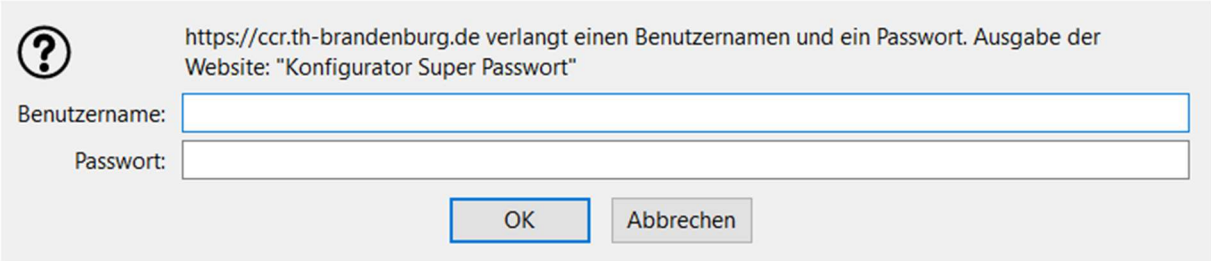


Abbildung 4: Zugangsbeschränkung per "htaccess"

3.2 Backend Webinterface

Die benutzerfreundliche Bedienung des Webinterfaces wird im Folgenden anhand mehrerer Darstellungen näher erläutert.

3.2.1 Auswahl von Standards und Objekten

Nach Aufruf des Backends werden über eine Dropdown-Liste die einzelnen Standardversionen aufgelistet. Durch Anklicken kann die zu bearbeitende Version ausgewählt werden. Nach Wahl eines Standards werden in einer weiteren Dropdown-Liste die zum Standard zugehörigen Objekte angezeigt.

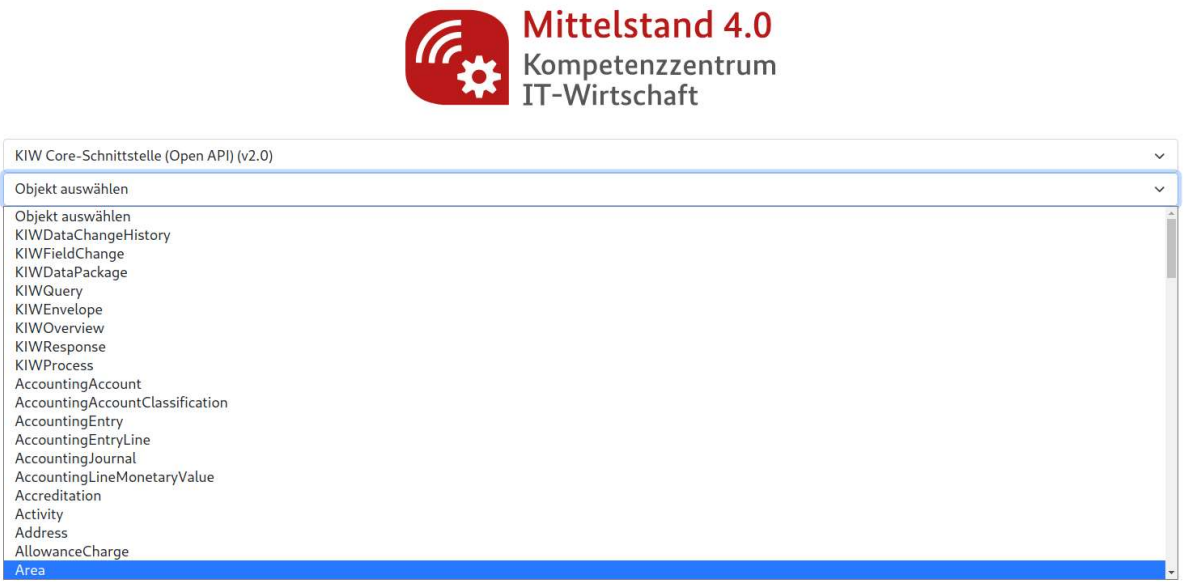


Abbildung 5: Auswahl des Standards und die Auflistung der zugehörigen Objekte

Abbildung 6: Auswahl Standard und der Button zum Hinzufügen neuer Objekte

3.2.2 Hinzufügen von Objekten

Zu jedem Standard lassen sich weitere Objekte durch einen Button (siehe Abbildung 6) hinzufügen. Dafür wird nach Betätigung des Buttons ein Popup Fenster geöffnet, durch welches die Objektdaten eingetragen werden können. Hierbei muss, wie in der Abbildung 7 dargestellt, mindestens der Name des neuen Objekts angegeben werden. Das Formular wird hierbei auf die entsprechende Korrektheit der Eingaben überprüft. Der neue Objekteintrag wird erst nach korrekter Eingabe gestattet.

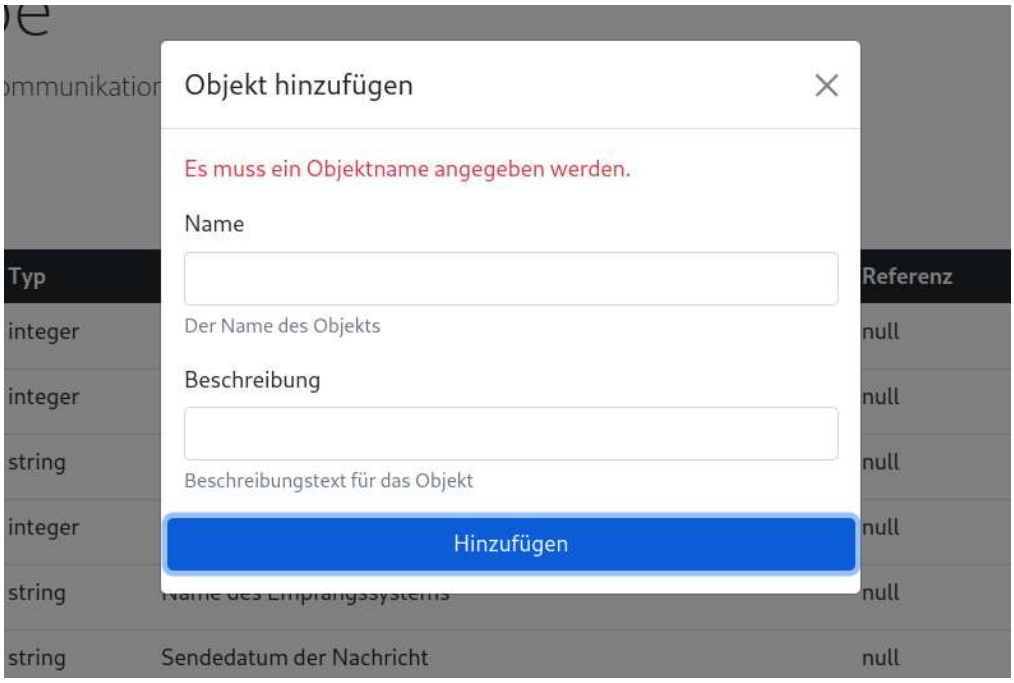


Abbildung 7: Formular für das Hinzufügen neuer Objekte.

3.2.3 Auflistung aller Attribute eines ausgewählten Objekts

Nach Auswahl eines bereits vorhandenen Objekts oder nach dem Hinzufügen eines neuen, wird das jeweilige Objekt in einer erweiterten Ansicht mit allen Attributen angezeigt. Die zum Objekt zugehörigen Attribute werden tabellarisch mit ihren entsprechenden Eigenschaften aufgelistet. Dazu zählen der Name, der Datentyp, eine Kurzbeschreibung, sowie einer Objektreferenz (falls das Attribut vom Typ „object“ ist). In dieser Ansicht lassen sich durch weitere Buttons Attribute hinzufügen, bearbeiten oder löschen. Das ausgewählte Objekt lässt sich ebenfalls über Buttons bearbeiten und löschen.

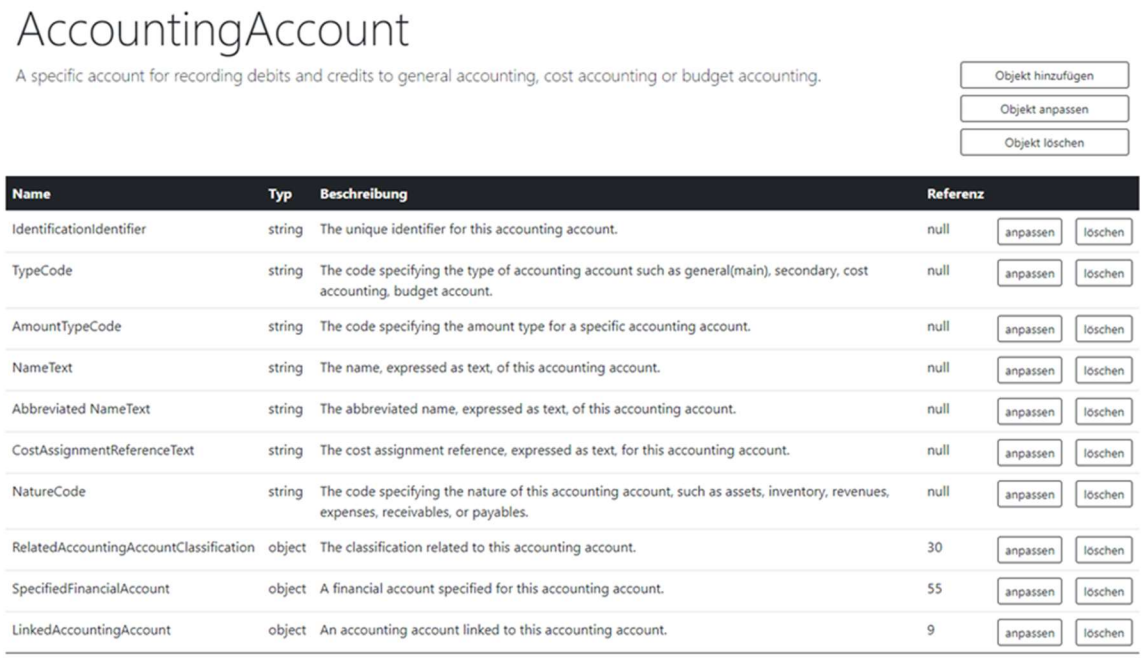


Abbildung 8: Erweiterte Objektansicht mit allen Attributen.

3.2.4 Hinzufügen von Attributen

Analog zum Hinzufügen eines neuen Objekts, ist das Hinzufügen weiterer Attribute sowie die Bearbeitung der Attribute/Objekte ebenfalls über Popup Formulare realisiert.

Beim Hinzufügen neuer Attribute müssen ein Name und Objekttyp angegeben werden. Der Objekttyp lässt sich aus einer Liste unterstützter Objekttypen wählen. Falls für das Attribut der Typ „object“ ausgewählt wird, so wird eine Objektreferenz verlangt, welche aus der Referenzliste ausgewählt werden kann. Damit wird sichergestellt, dass nur vorhandene Objekte referenziert werden können. Auch hierbei wird das Formular auf entsprechende Korrektheit der Angaben hin überprüft.

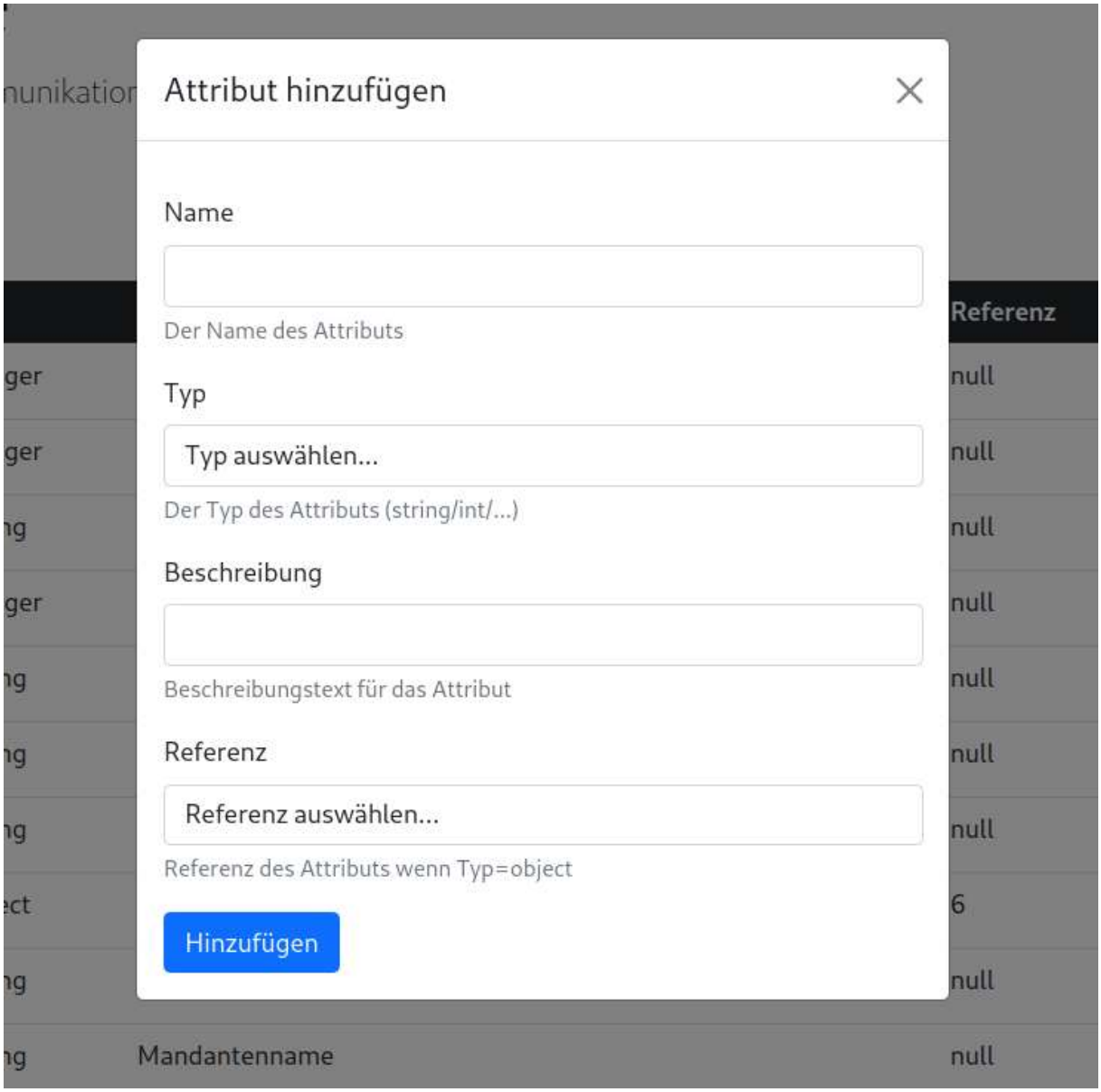


Abbildung 9: Formular für das Hinzufügen neuer Attribute.

Abbildung 10: Formular wirft bei fehlerhafter Eingabe eine Meldung.

3.2.5 Löschen von Attributen

Für das Löschen von Attributen/Objekten wird eine Bestätigungsabfrage erzeugt. Der Benutzer muss hierbei erst mit „Ja“ bestätigen, um den Löschvorgang einzuleiten. Bei Objekten wird vor dem Löschvorgang noch überprüft, ob das zu löschende Objekt Abhängigkeiten zu anderen Attributen besitzt. Damit soll sichergestellt werden, dass keine Objekte gelöscht werden können, auf die von Attributen referenziert wurde.

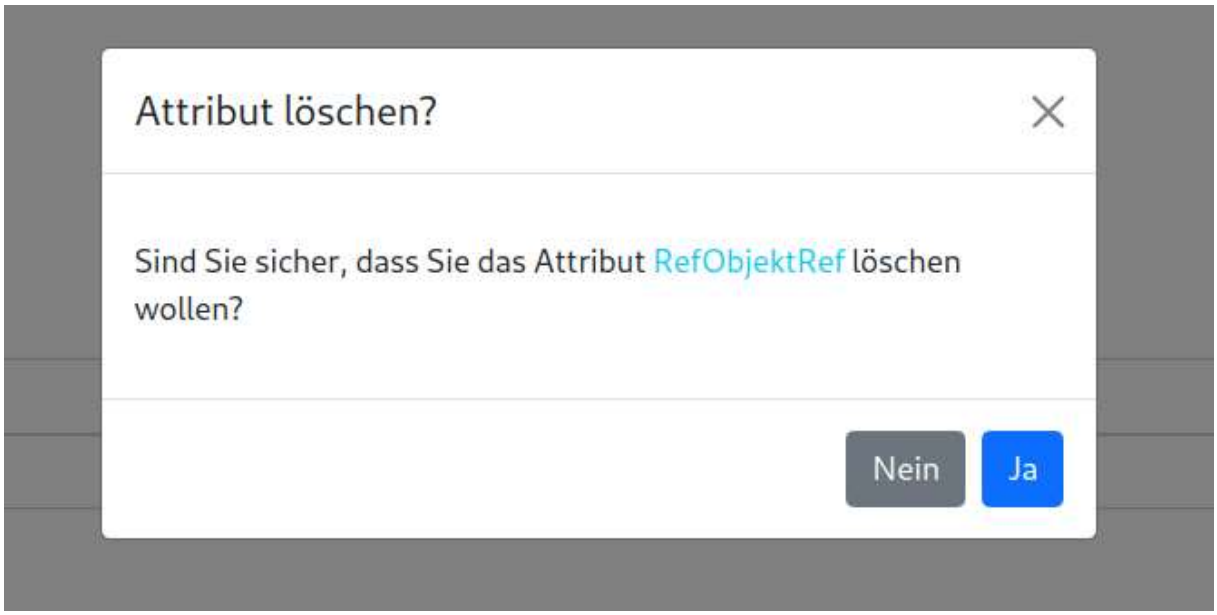


Abbildung 11: Sicherheitsabfrage zum Löschen eines Attributs.

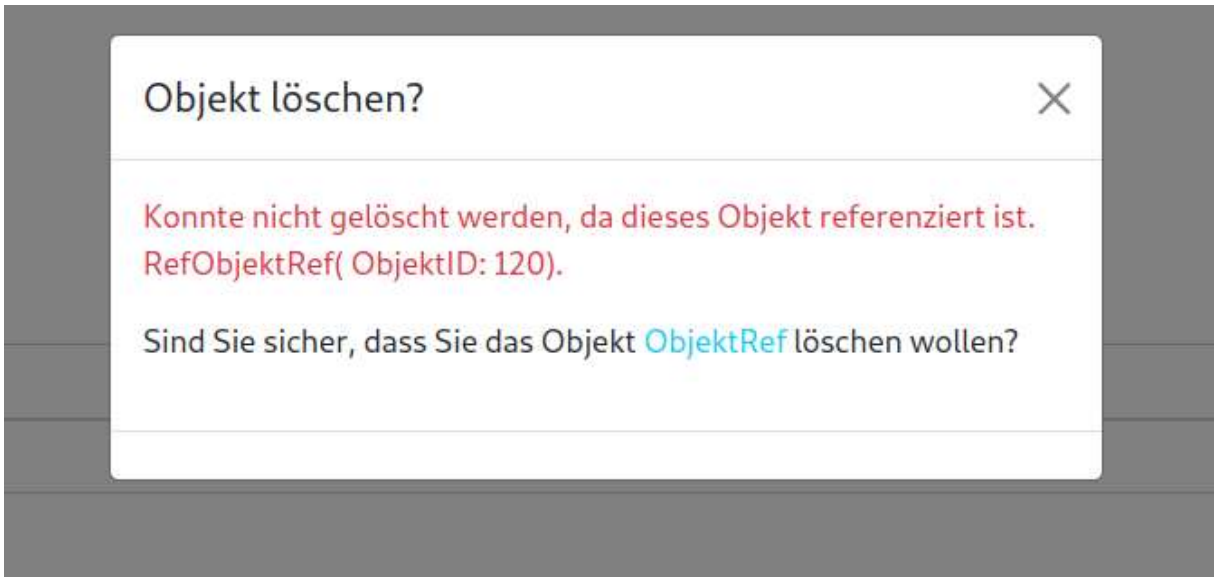


Abbildung 12: Sicherheitsabfrage zum Löschen eines Objekts. Löschvorgang verwiesen da eine direkte Abhängigkeit zu einem Attribut besteht.

Nach dem Lösch-/Bearbeitungsvorgängen wird das Webinterface mit den entsprechenden Änderungen aktualisiert. Beim Hinzufügen/Bearbeiten/Löschen von Attributen wird dabei die Attributen Tabelle mit den aktualisierten Einträgen erneuert.

3.3 Exporter

Der Exporter interpretiert das ausgefüllte Formular der Konfigurator Seite und erstellt daraus eine OpenAPI- und KIW-Schnittstellenkatalog- valide YAML-Datei zum Downloaden. Umgesetzt ist der Exporter in PHP.

Die Exportfunktion ist in dieser Version implementiert worden. Die eigene Auswahl an Objekten über das Konfigurator-Frontend, lässt sich nun über den Export-Button als OpenAPI-valider Objektauszug exportieren. Dabei werden die ausgewählten Objekte mittels eines AJAX POST Request an den Exporter Endpunkt kommuniziert. Dieser interpretiert dann die ausgewählten Objekte und baut daraus eine valide YAML nach der OpenAPI Spezifikation. Die gesamte YAML Datei besteht hierbei aus zwei Komponenten, dem statischen und dem dynamischen Teil. Der dynamische Teil besteht aus sich ändernden Informationen, abhängig von der Benutzereingabe. Dazu zählen beispielsweise die Objekte, oder die Version des ausgewählten Standards. Der statische Teil hingegen, enthält konstante Informationen, welche in absehbarer Zeit nicht geändert werden, z.B. im „info“ Abschnitt der YAML: Kontaktdaten des KIW.

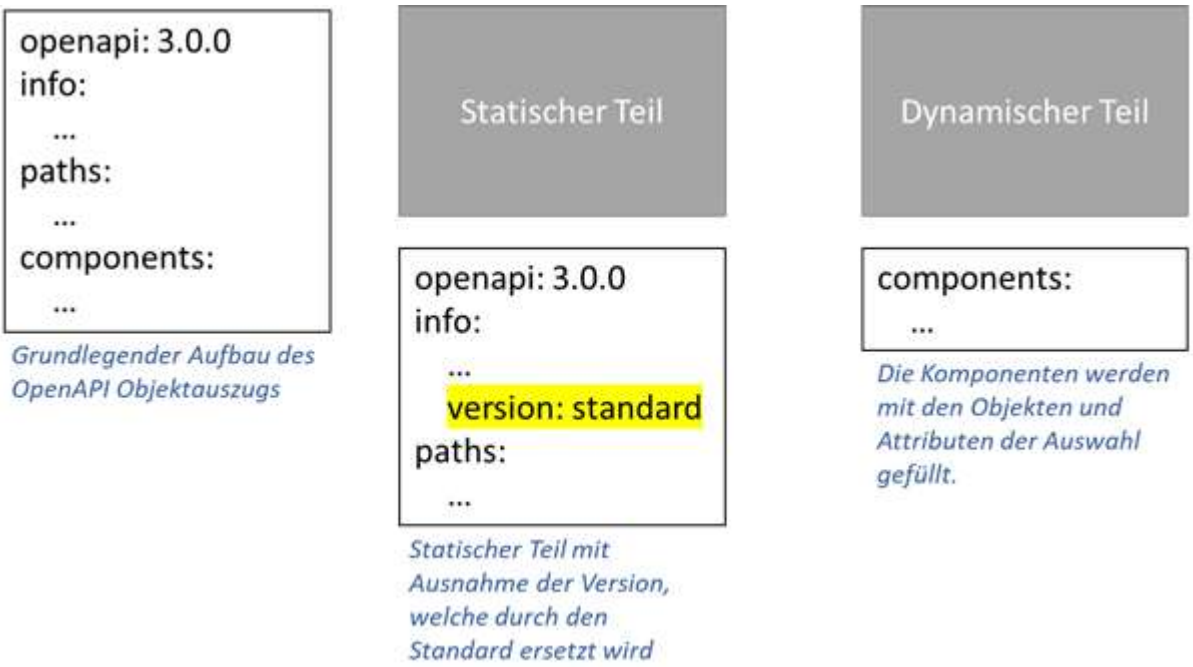


Abbildung 13 Veranschaulichung statischer/dynamischer Teil

Die dynamischen Daten sind initial im JSON Format gespeichert, da diese vom Frontend in dieser Form an den Exporter Endpunkt übermittelt werden. Der gesamte Exporter verwendet die in PHP standardisierte „yaml_emit“ Funktion um mit diesen Daten den OpenAPI Objektauszug im YAML Format zu erstellen. Voraussetzung für diesen Prozess ist daher eine geeignete PHP Extension, welche die standardisierte „yaml_emit“ Funktion bereitstellt.

Der gesamte YAML Objektauszug wird nach Beendigung der Umwandlung, zurück an das Frontend geschickt, wo es dann als Download zur Verfügung gestellt wird.

4 Aktualisierte Komponenten

Im dritten Kapitel “Aktualisierte Komponenten” werden die Änderungen an bestehenden Komponenten seit der letzten Prototypenversion beschrieben.

4.1 Importer

Der Importer ermöglicht es Objektbibliotheken die als Beschreibungssprachen vorliegen in die Konfigurator Datenbank zu importieren. Dieser Prozess ist in drei Schritte aufgeteilt.

Im ersten Schritt wird die Objektbibliothek über die Skriptsprache (JS) eingelesen. Dabei wird eine Analyse gestartet, um die Beschreibungssprache zu ermitteln.

Nach erfolgreicher Erkennung der Sprache wird in Schritt zwei das benötigte Beschreibungssprachen Modul geladen.

Im dritten und letzten Schritt werden mit Hilfe des geladenen Moduls die Objekte, Attribute und deren Eigenschaften extrahiert und danach in die Konfigurator Datenbank geschrieben.

Der Importer hat sich in seiner grundlegenden Funktionalität nicht geändert. Es wurde das Format für die zu importierenden OpenAPI Objektauszüge angepasst. Die „import.yaml“ wurde somit auf einen Standard vereinheitlicht. Folgernd wird beim Export und Import der Auszüge, bei der Referenz eines Objekts syntaktisch nicht mehr das „allOf“ Stichwort verwendet. Die Referenz soll ab jetzt für Objekttypen direkt als Element der „items“ Liste angehängt werden.

```
Process:
  type: object
  description: Objekt für die Beschreibung der angewendeten Methode.
  $ref: '#/components/schemas/KIWProcess'
```

Abbildung 14 Syntaktische Veränderung ‘$ref’ bei der import.yaml

4.2 Datenbank

Die Struktur der Datenbank ist seitdem letzten Prototypen unverändert geblieben. Alle neuen Schnittstellenfunktionen sind mit der aktuellen Datenbankstruktur integrierbar gewesen. Es ließen sich alle notwendigen Daten abrufen. Besonders wichtige Datenbankaufrufe sind im Bereich des Backends realisiert worden. Dabei sind Vorgänge zum Hinzufügen, Bearbeiten und Löschen von Einträgen in den Tabellen „Object“ und „Attribut“ über das Webinterface hinzugekommen.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	Name	varchar(255)	utf8_bin		No	*None*		
3	Type	varchar(45)	utf8_bin		No	*None*		
4	Cardinality	char(1)	utf8_bin		No	o		
5	Description	tinytext	utf8_bin		Yes	*NULL*		
6	ObjectID	int(11)			No	*None*		
7	ReferenceID	int(11)			Yes	*NULL*		

Abbildung 15: Tabelle „Attribut“ mit entsprechenden Spalten.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	id	int(10)		UNSIGNED	No	*None*		AUTO_INCREMENT
2	name	varchar(250)	utf8mb4_unicode_ci		No	*None*		
3	standardid	int(10)			No	*None*		

Abbildung 16: Tabelle „Gruppen“ mit entsprechenden Spalten.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	Name	varchar(255)	utf8_bin		No	*None*		
3	Description	tinytext	utf8_bin		Yes	*NULL*		
4	StandardID	int(11)			No	*None*		

Abbildung 17: Tabelle „Object“ mit entsprechenden Spalten.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	ID	int(11)			No	*None*		AUTO_INCREMENT
2	Standard	varchar(255)	utf8_bin		No	*None*		
3	Version	varchar(255)	utf8_bin		No	*None*		

Abbildung 18: Tabelle „Standard“ mit entsprechenden Spalten.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	gruppenid	int(10)			No	*None*		
2	attributid	int(10)			No	*None*		

Abbildung 19: Tabelle „Vorauswahl“ mit entsprechenden Spalten.

4.3 Frontend

Über das Web-Frontend interagiert der Anwender mit dem Konfigurator. Das Frontend ist mit den Web-Sprachen HTML, JSON, CSS, JS und PHP umgesetzt. Als Frameworks kommen Bootstrap, jQuery und Popper zum Einsatz. Das Frontend gliedert sich in zwei Seiten. Ruft der Nutzer den Konfigurator auf, befindet er sich auf der Landingpage.

Auf der Landingpage werden die zur Verfügung stehenden Versionen des Schnittstellenkatalogs zur Auswahl aufgelistet. Weiterhin ist es hier möglich vorkonfigurierte Beispiele, die so genannten Referenzmodelle, auszuwählen.

Auf der Konfigurator Seite ist es möglich die ausgewählte Version oder Referenzmodell nach den eigenen Vorstellungen zu konfigurieren. Dazu können Objekte und Attribute der eigenen Auswahl hinzugefügt oder abgewählt werden. Das Tool achtet dabei darauf das vorhandene Abhängigkeiten zwischen Objekten und Attributen eingehalten werden.

Das Frontend ist größtenteils unverändert geblieben, da die grundlegende Benutzerinteraktion bereits implementiert war. Jedoch wurde nun die Export-Funktionalität implementiert, sodass bei Betätigung des Buttons die exportierte Auswahl im YAML Format als Download zur Verfügung gestellt wird.

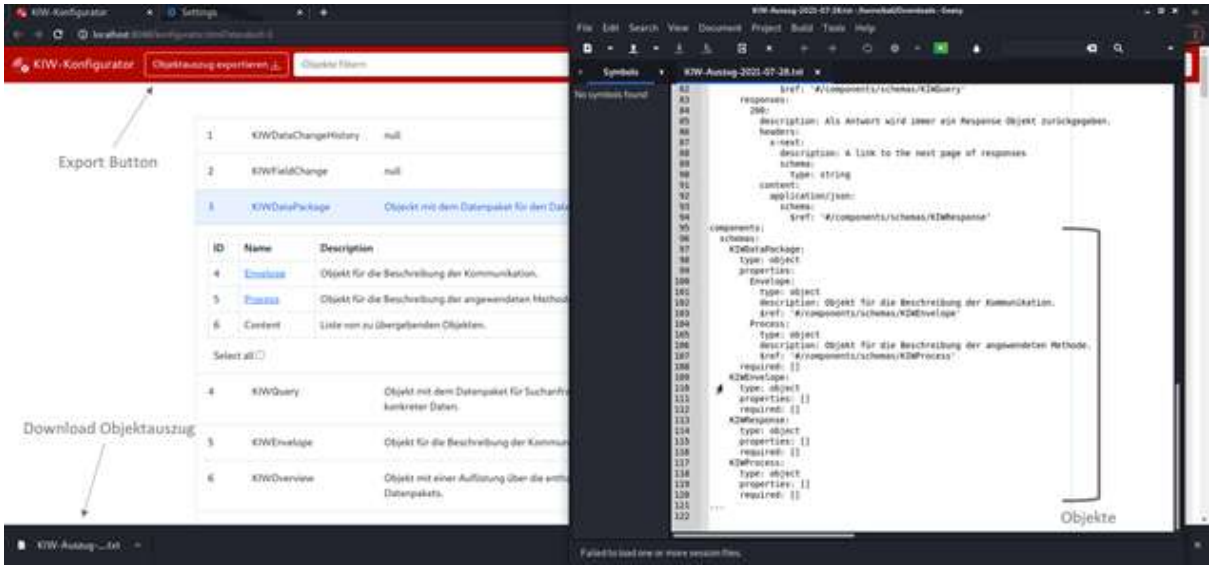


Abbildung 20: Prozess des Exportierens und Veranschaulichung des Inhalts

Zusätzlich wurde das Frontend im Sinne des Responsive Webdesign angepasst. Der Konfigurator lässt sich nun auch über mobile Endgeräte (Smartphones, Tablets) korrekt darstellen und nutzen. Je nach Bildschirmgröße werden einige Elemente zur Benutzerfreundlichkeit ausgeblendet. Diese beeinträchtigen jedoch nicht die Grundfunktionalität: dem korrekten Export der ausgewählten Objekte/Attribute.

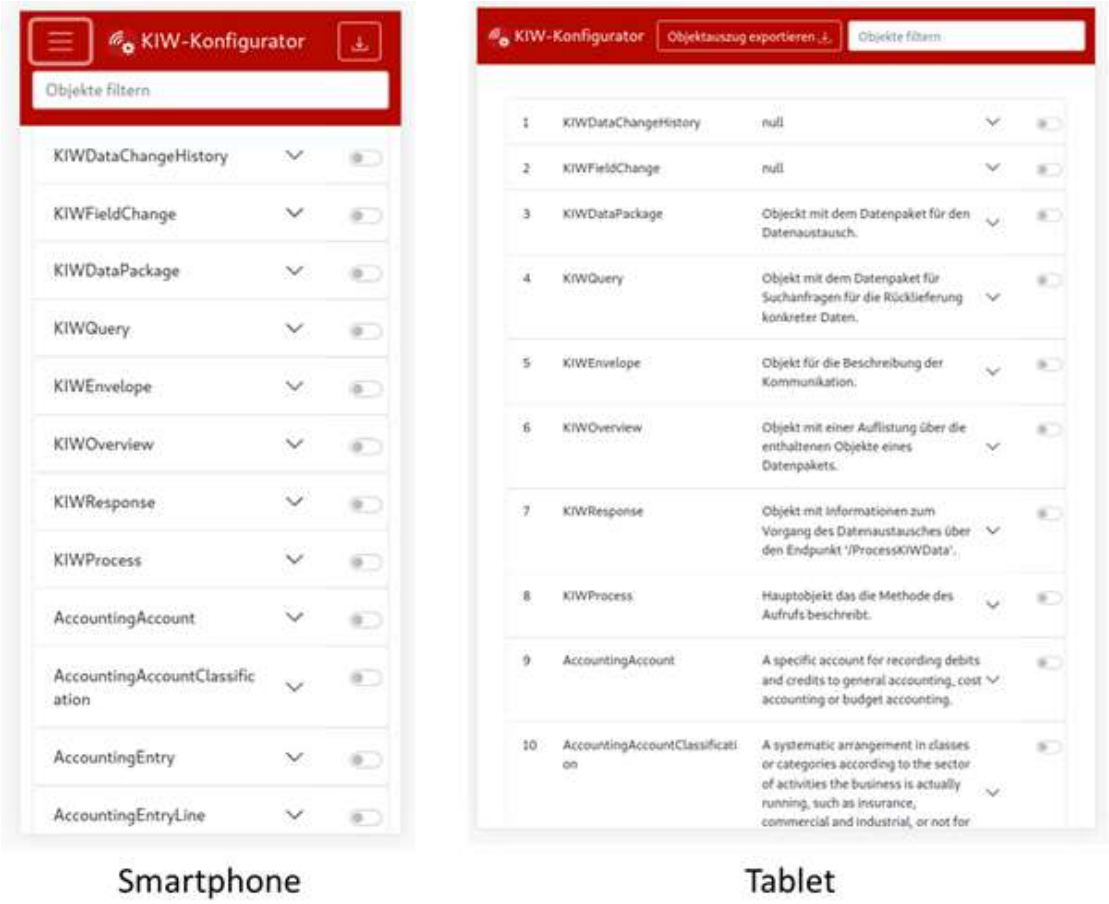


Abbildung 21: „Responsive Webdesign“ des Konfigurators am Beispiel von Smartphone und Tablet

4.4 Schnittstelle

Die Schnittstelle bildet für die Clientseite den Kontaktpunkt zur Datenbank. Damit wird der sichere Datenaustausch mit der Datenbank gewährleistet. Diese wurde über einen PHP-Endpunkt realisiert. Die sichere Authentifizierung findet dabei über diesen Endpunkt statt. Über den Endpunkt werden die Anfragen an die Datenbank weitergeleitet und die Antwort im JSON Format zurück an den Client gereicht. Die Schnittstelle wurde, um die für das Backend notwendigen Funktionalitäten erweitert. Dabei wurden zusätzlich zu den Leseoperationen, folgende weitere Funktionen bereitgestellt, welche im Webinterface verwendet werden:

- Löschen der Objekte/Attribute
- Bearbeiten der Objekte/Attribute
- Hinzufügen der Objekte/Attribute
- Abfrage abhängiger Attribute zu einem Objekt
- Abfrage aktueller Objekt-/Attributdaten

5 Interner Testzugang

Im vierten Kapitel “Interner Testzugang” sind Zugangsdaten festgehalten, die verwendet werden können um den aktuellen Stand des Konfigurators zu testen, und so die Möglichkeit zu haben Feedback für weitere Versionen geben zu können.

Zugang:

URL	https://api-konfigurator.th-brandenburg.de
Benutzername	konfigurator
Passwort	kiw-thb

6 Abbildungsverzeichnis

Abbildung 1: Wachstum an Web-APIs aus dem öffentlichen Verzeichnis für Web-APIs von Programmableweb.com 3
Abbildung 2: Ablaufprozess des Konfigurators, vom Import bis zum Export. 5
Abbildung 3: Ablaufprozess des Konfigurators aus technischer Sicht, vom Import bis zum Export. 5
Abbildung 4: Zugangsbeschränkung per "htaccess" 6
Abbildung 5: Auswahl des Standards und die Auflistung der zugehörigen Objekte 7
Abbildung 6: Auswahl Standard und der Button zum Hinzufügen neuer Objekte. 7
Abbildung 7: Formular für das Hinzufügen neuer Objekte. 8
Abbildung 8: Erweiterte Objektansicht mit allen Attributen. 8
Abbildung 9: Formular für das Hinzufügen neuer Attribute. 9
Abbildung 10: Formular wirft bei fehlerhafter Eingabe eine Meldung. 10
Abbildung 11: Sicherheitsabfrage zum Löschen eines Attributs. 10
Abbildung 12: Sicherheitsabfrage zum Löschen eines Objekts. Löschvorgang verwiesen da eine direkte Abhängigkeit zu einem Attribut besteht. 11
Abbildung 13 Veranschaulichung statischer/dynamischer Teil 12
Abbildung 14 Syntaktische Veränderung ‘$ref’ bei der import.yaml 13
Abbildung 15: Tabelle „Attribut“ mit entsprechenden Spalten. 14
Abbildung 16: Tabelle „Gruppen“ mit entsprechenden Spalten. 14
Abbildung 17: Tabelle „Object“ mit entsprechenden Spalten. 14
Abbildung 18: Tabelle „Standard“ mit entsprechenden Spalten. 14
Abbildung 19: Tabelle „Vorauswahl“ mit entsprechenden Spalten. 15
Abbildung 20: Prozess des Exportierens und Veranschaulichung des Inhalts 15
Abbildung 21: „Responsive Webdesign“ des Konfigurators am Beispiel von Smartphone und Tablet 16

Literaturverzeichnis

[1] S. Sindermann , Modellbasierte virtuelle Produktentwicklung, Heidelberg : Springer-Verlag Berlin, 2014.

[2] OpenJS Foundation, „Node.js,“ OpenJS Foundation, 02 June 2021. [Online]. Available: https://nodejs.org/. [Zugriff am 02 June 2021].

7 Kontakt

Wenn Sie Fragen oder Anregungen zum zweiten Prototyp Schnittstellen- Konfigurator haben, lassen Sie es uns wissen. Wir stehen Ihnen gerne zur Verfügung.

Ihre Ansprechpartner:

Prof. Dr. Andreas Johannsen
Telefon: +49 3381 355 - 256
Email: andreas.johannsen@itwirtschaft.de

Matthias Dobkowicz
Telefon: +49 3381 355- 864
Email: matthias.dobkowicz@itwirtschaft.de

Maximilian Müller
Telefon: +49 3381 355 -870
Email: maximilian.mueller@itwirtschaft.de

Kompetenzzentrum IT-Wirtschaft vertreten durch:

Bundesverband IT-Mittelstand e.V. (BITMi)
Hauptstadtbüro Berlin: Haus der Bundespressekonferenz

Schiffbauerdamm 40, 10117 Berlin T +49 30 22605 005

www.itwirtschaft.de

Was ist Mittelstand-Digital?

Das Mittelstand 4.0-Kompetenzzentrum IT-Wirtschaft ist Teil der Förderinitiative Mittelstand-Digital. Mittelstand-Digital informiert kleine und mittlere Unternehmen über die Chancen und Herausforderungen der Digitalisierung. Regionale Kompetenzzentren helfen vor Ort dem kleinen Einzelhändler genauso wie dem größeren Produktionsbetrieb mit Expertenwissen, Demonstrationszentren, Netzwerken zum Erfahrungsaustausch und praktischen Beispielen. Das Bundesministerium für Wirtschaft und Klimaschutz ermöglicht die kostenfreie Nutzung aller Angebote von Mittelstand-Digital.

Weitere Informationen finden Sie unter:
www.mittelstand-digital.de

Impressum

Konzeption und Text: Prof. Dr. Andreas Johannsen, Kompetenzzentrum IT-Wirtschaft,

Bildnachweis: Unsplash

Stand: Oktober 2021