

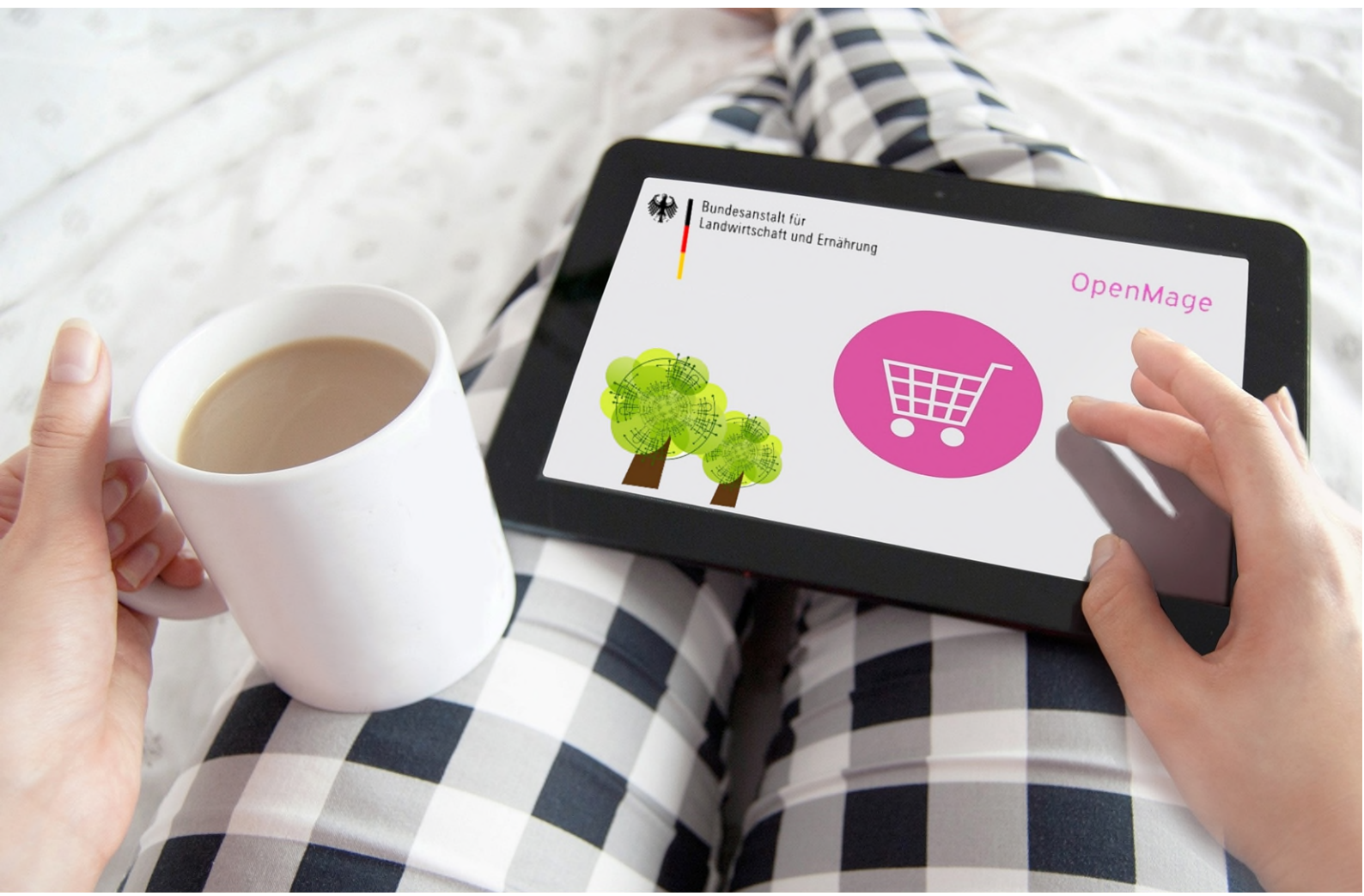


Bundesanstalt für
Landwirtschaft und Ernährung

Anhang O – Schnittstellenbeschreibung

BLE-Online Shop (Medienservice)

Version: 4.1
Status: Abgenommen
Revision: 12.02.2026



Inhalt

1.	Dokumentenhistorie.....	3
2.	Einleitung Test- und Echt-Systemverbund	4
2.1.	Artikelstamm.....	5
2.2.	Lagerbestand.....	5
2.3.	Lagerfreigaben	5
2.4.	Bestellungen.....	6
2.5.	Nachbesserungen durch Nachlieferung.....	6
3.	Allgemeines	6
4.	Artikelstamm.....	7
4.1.	Funktion: Gesamter Artikelstamm	7
4.2.	Funktion: Einzelner Artikelstamm.....	11
5.	Lagerbestand.....	16
5.1.	Funktion: Lagerfreigabeauftrag	16
5.2.	Funktion: Gesamter Lagerbestand	22
5.3.	Funktion: Einzelner Lagerbestand.....	29
6.	Bestellungen.....	36
6.1.	Funktion: Einzelne Bestellung.....	36
6.2.	Funktion: Lieferstatus einzelner Bestellung inkl. Nachsendungen	46
7.	Abbildungsverzeichnis	49
8.	Tabellenverzeichnis	50

1. Dokumentenhistorie

Datum	Version	Änderungsgrund	Bearbeiter
11.05.2023	0.0	Initial	Marco Junker Holger Kögel
17.05.2023	0.1	Umstrukturierung und REST	Frank Rochlitzer
24.05.2023	0.9	Artikelstammdaten REST	Frank Rochlitzer
25.05.2023	0.9.1	Lagerbestand und Bestellungen REST	Frank Rochlitzer, Holger Kögel
31.05.2023	0.9.2	Review und Anpassungen	Frank Rochlitzer
21.06.2023	0.9.3	Anpassungen, Fragen/Kommentare	Marco Junker
03.07.2023	0.9.5	Anpassungen, Fragen/Kommentare	Frank Rochlitzer
04.07.2023	0.9.8	Anpassungen, Fragen/Kommentare	Marco Junker
13.07.2023	0.9.11	Anpassungen, Kommentare	Hans Mackowiak
14.07.2023	1.0	Abnahme	Marco Junker
28.09.2023	1.1	Datumsflag für Lagerbewegung	Holger Kögel
23.01.2025	1.5	Erweiterung	Marco Junker Holger Kögel Eugenie Kibke
12.08.2025	1.7	Abnahme	Marco Junker
09.10.2025	2.0	Überarbeitung und Abnahme	Marco Junker Holger Kögel
12.01.2026	3.0	Überarbeitung und Abnahme	Marco Junker Holger Kögel
11.02.2026	4.0	Überarbeitung und Abnahme, Fehlerbehebung, Zeitstempel	Eugenie Kibke Holger Kögel Marco Junker
12.02.2026	4.1	Fehlerkorrektur 6.2 bzgl. Vollständigkeit Attributbeschreibungen und Abbildung in den JSON sowie kleinere AN-Verweise	Marco Junker

2. Einleitung Test- und Echt-Systemverbund

Grundlegend gilt es, nur relevante Datensätze und Attribute (Felder) zur Aufgabenerfüllung zwischen den Systemen BLE-Online Shop (BLE-Medienservice) und dem System des Vertriebsdienstleisters - in beide Richtungen - via API (REST, SOAP) auszutauschen.

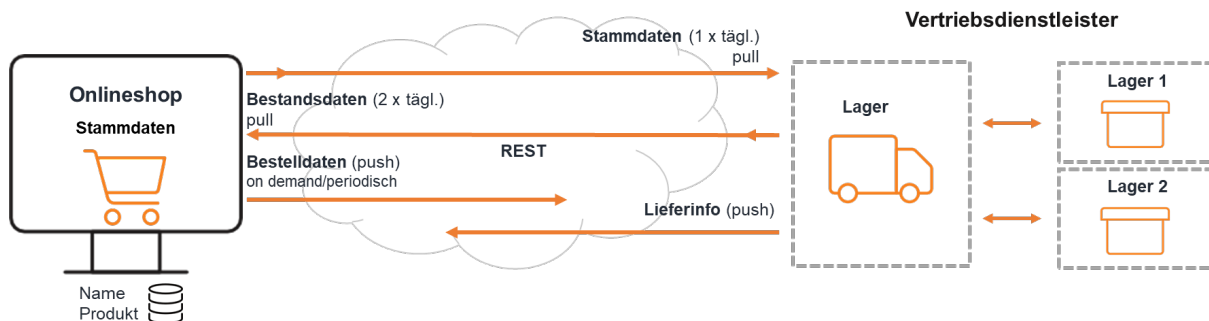


Abbildung 1: Kommunikation zwischen BLE-Online Shop (Auftraggeber - AG) und Vertriebsdienstleister (Auftragnehmer - AN)

Hierfür gilt es, **zwei** unabhängige Instanzen/Systemkombinationen zu etablieren:

- Testsystem: Test-Online Shop (BLE)
angebunden an das
Test-WWS (Vertriebsdienstleister)
- Echtsystem: Echt-Online Shop (BLE)
angebunden an das
Echt-WWS (Vertriebsdienstleister)



Demnach muss die Möglichkeit gegeben sein, dass Entwicklungen und Anpassungen zunächst auf dem Testsystem erprobt werden, bevor diese auf dem Echtsystem angewendet werden.

- Das Testsystem beinhaltet fiktive / nicht reale Daten und Bestände.
- Das Echtsystem hingegen beinhaltet reale Daten und Bestände.

2.1. Artikelstamm

Für **Artikelstammdaten** gilt der Austausch und die Synchronhaltung wie unter anderem

- grundlegende Artikelinfos (Produkt ID, Artikelnummer, Titel, Auflage, Mandant, Referat,...)
- mandantenzugehöriger Informationen

Es werden somit nur Stammdaten mit dem **Flag „in_Sale“** (früher „ImVertrieb“) für rein physikalisch bei dem Vertriebsdienstleister lagernde Artikel ausgetauscht.

Eindeutiges Attribut/Relations-ID/Key des Artikels ist [entity_id] = Product ID

2.2. Lagerbestand

Für **Lagerbestandsdaten** gilt der Austausch und die Synchronhaltung

- einzelner Lagerbestände aller Lagerplätze in Stück (Anlieferung, Haupt, Sperr, Makulierung Papier, Makulierung Rest) je Artikeldatensatz (Auflage)

Eindeutiges Attribut/Relations-ID/Key des Artikels ist [entity_id] = Product ID

Erläuterungsbeispiel für die Relation:

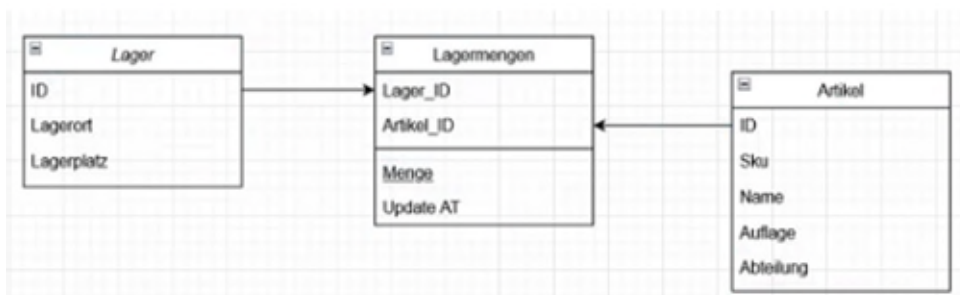


Abbildung 2: Erläuterungsbeispiel Relation ProductID

2.3. Lagerfreigaben

Für die **Lagerfreigabe** gilt die Übersendung der Anweisung an den Vertriebsdienstleister, wie für einzelne Artikel (Auflagen) zu verfahren ist bzgl. Lagerbestandsbuchungen und Verwendung

- bei Nachdrucken (ein Datensatz = gleiche Auflage)
- bei neuen Auflagen (zwei Datensätze = zwei Auflagen)

Eindeutiges Attribut/Relations-ID/Key des Artikels ist [entity_id] = Product ID

2.4. Bestellungen

Für **Bestelldaten** und zur Abwicklung der Lieferung durch den Vertriebsdienstleister gilt die Überlieferung von

- personenbezogenen Daten (Kundennummer, Name, Firma, Lieferanschrift, Kontaktangaben für Lieferabstimmungen/Spedition)
- Bestellinformationen (Bestellnummer, Datum)
- Positionsinformationen zu den bestellten Artikeln (Artikel, Menge)
- Rechnungsbetrag (Bruttopreis in EUR, Nettopreis in EUR, MwSt.-Anteil in EUR)
- Warenwertinformationen in EUR für die Ausfuhr in Drittländer

Für **Informationen zur Lieferung** gilt die Übersendung von z.B.

- bestellungsabhängigen Lieferinformationen (Liefernummer, Lieferschein PDF, Transportdienstleister, Sendungsnummer, URL)

Es werden demnach nur Bestellungen übermittelt, die physikalisch durch den Vertriebsdienstleister abzuwickeln sind.

2.5. Nachbesserungen durch Nachlieferung

Für **Informationen der Nachlieferung** zur Liefermangelbeseitigung gilt die Übersendung von

- bestellungsabhängigen Lieferinformationen (Liefernummer, Lieferschein PDF, Transportdienstleister, Sendungsnummer, URL)

3. Allgemeines

Die Authentifizierung zur Nutzung der Schnittstelle findet per OAuth Version 1 statt. Für den Online Shop erfolgt die Konfiguration über das Backend des Online Shops.

Folgende Endpunkte zur OAuth-Authentifizierung stehen zur Verfügung:

- [my.domain.local]/oauth/initiate - Endpunkt zum Abruf des Request Tokens
- [my.domain.local]/admin/oauth_authorize - Endpunkt für Benutzerautorisierung (Admin).
- [my.domain.local]/oauth/token - Endpunkt zum Abruf des Zugangstokens

Zuordnung von HTTP-Methoden zu klassischen CRUD-Verben:

- GET → OPERATION RETRIEVE,
- POST → OPERATION CREATE,
- PUT → OPERATION UPDATE,
- DELETE → OPERATION DELETE

4. Artikelstamm

4.1. Funktion: Gesamter Artikelstamm

Beschreibung:	Der BLE-Online Shop (Medienservice) verwaltet alle Artikelstammdatensätze. Um eine tägliche Plausibilität der Artikelstammdatenverwaltung beider Systeme gewährleisten zu können, gilt es, einmal täglich einen gesamten Abgleich auf Veränderungen (neue Artikel, Veränderung bestehender Artikel) durchzuführen. Hierzu zieht sich der Vertriebsdienstleister die gesamten Daten vom BLE-Online Shop (pull), prüft auf Abweichungen und verarbeitet Abweichungen für eine synchrone Datenhaltung.
Weg:	Vertriebsdienstleister ← BLE-Online Shop
Aktivität:	Vertriebsdienstleister zieht alle Daten (pull)
Übertragungsart:	REST, Liste
Übertragungshäufigkeit:	Täglich, 3:00 Uhr
Prozess:	1. Vertriebsdienstleister zieht Daten 2. prüft auf Abweichungen (Plausibilität aller Felder) 3. verarbeitete Abweichungen anhand eindeutiger [ShopProduktID]
Konsequenz:	- o Wenn [ShopProduktID] unbekannt = neuer Artikel Datensatz anlegen - o Wenn [ShopProduktID] bekannt = Update Artikel Datensatz

Tabelle 1: Gesamter Artikelstamm - Übersicht

Bezeichnung	Feld	Typ	Pflicht	Erläuterung
ShopProduktID	entitiy_id	Int unsigned	Ja	Eindeutige Zuweisungsnummer des Datensatz/Artikels Herkunft: [entity_id] Product ID aus Artikelstamm (BLE-Online Shop)
Mandant	client	String (255)	Ja	Mandantenzugehörigkeit des Datensatz/Artikels Herkunft: division_id [parentname] Mandant aus Fachbereich (BLE-Online Shop)
Artikelnummer	origin_sku	String (255)	Ja	Artikelnummer des Datensatz/Artikels

				Herkunft: [origin_sku] Basisartikelnummer aus Artikelstamm (BLE-Online Shop)
Titel	name	Text	Ja	Titelname des Datensatz/Artikels Herkunft: [name] aus Artikelstamm (BLE-Online Shop)
MediumTraeger	medientraeger	String (255)	Ja	Medienträger des Datensatz/Artikels Herkunft: [medientraeger] aus Artikelstamm (BLE-Online Shop)
MediumTyp	medientype	String (255)	Ja	Medientyp des Datensatz/Artikels Herkunft: [medientyp] aus Artikelstamm (BLE-Online Shop)
Auflage	auflage	String (255)	Ja	Auflagenart des Datensatz/Artikels Herkunft: [auflage] aus Artikelstamm (BLE-Online Shop)
Referat	devision	String (255)	Ja	Ausgabewert des Referats / Referatsbezeichnung Herkunft: [devision] aus Mandant aus Fachbereich (BLE-Online Shop)

Tabelle 2: Gesamter Artikelstamm - Attributdarstellung

[AN 1] Es müssen die folgenden REST-Aufrufe implementiert werden:

GET [my.domain.local]/api/rest/vdl_products – Artikelstammdatenabfrage

[AN 2] Der REST-Request /api/rest/vdl_products zur Abfrage von Artikelstammdaten, hat folgenden Aufbau:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
URL Parameter			
limit	int	Default: 10 Maximum: 100	
page	int	Seitenzahl, für die Ergebnisse zurückgegeben werden sollen	

Tabelle 3: Gesamter Artikelstamm - Rest-Request

[AN 3] Der REST-Response aus [AN 2] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	<pre>{ "entity_id": int : { "entity_id": int, "client": string, "origin_sku": string, "name": string, "medientraeger": string, "medientype": string, "auflage": string, "division": string } }</pre>	x

Tabelle 4: Gesamter Artikelstamm - REST Response bei erfolgreicher Artikelstammapfrage

Beispiel: [my.domain.local]/api/rest/vdl_products?limit=100&page=2

```
{
  „101“: {
    „entity_id“: „101“,
    "client": „BOEL“,
    „origin_sku“: „1234ABC“,
    "name": "Testartikel",
    "medientraeger": "download",
    "medientype": "typ",
    "auflage": "1",
    "division": "631"
  }
}
```

[AN 4] Der REST-Response aus [AN 2] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
error	Array	Enthält code & message	x
code	Integer	Fehlercode	

message	String	Fehlermeldung	
---------	--------	---------------	--

Tabelle 5: Gesamter Artikelstamm - REST Response bei fehlerhafter Artikelstammdatenabfrage

Beispiel:

```
{
  "error": [
    {
      "code": 400,
      "message": "Resource collection filtering error."
    }
  ]
}
```

Umgang im Fehlerfall:

- Von Seiten des AN ist ein Ticket an die Queue „Shop-System – Eingang“ zur weiteren Abstimmung und Klärung mit dem AG zu erstellen.
- Prüfung durch AG
- AG kann bei notwendigem Bedarf entsprechende Logs beim Betreiber des BLE-Online Shops anfordern

4.2. Funktion: Einzelner Artikelstamm

Beschreibung:	Der BLE-Online Shop verwaltet alle Artikelstammdatensätze. Nach aktiver Betätigung eines Schalters (Button) werden die Datensatzveränderung an den Vertriebsdienstleister gesendet. Dieser nimmt die Veränderung anhand eindeutiger ShopProduktID entgegen und verarbeitet diese.
Weg:	BLE-Online Shop → Vertriebsdienstleister
Aktivität:	BLE-Online Shop liefert einzelne Daten (push)
Übertragungsart:	SOAP, Einzeln
Übertragungshäufigkeit:	Nach aktiver Betätigung eines Schalters
Handeln:	1. BLE-Online Shop (Medienservice) sendet Daten 2. Vertriebsdienstleister nimmt Daten an 3. und verarbeitet diese anhand eindeutiger [ShopProduktID]
Konsequenz:	- o Wenn [ShopProduktID] unbekannt = neuer Artikeldatensatz anlegen - o Wenn [ShopProduktID] bekannt = Update Artikeldatensatz

Tabelle 6: Einzelner Artikelstamm - Übersicht

Bezeichnung	Feld	Typ	Pflicht	Erläuterung
ShopProduktID	entity_id	Int unsigned	Ja	Eindeutige Zuweisungsnummer des Datensatz/Artikels Herkunft: [entity_id] Product ID aus Artikelstamm (Online Shop)
Mandant	client	String (255)	Ja	Mandantenzugehörigkeit des Datensatz/Artikels Herkunft: division_id [parentname] Mandant aus Fachbereich (BLE-Online Shop)
Artikelnummer	origin_sku	String (255)	Ja	Artikelnummer des Datensatz/Artikels Herkunft: [origin_sku] Basisartikelnummer aus Artikelstamm (BLE-Online Shop)
Titel	name	Text	Ja	Titelname des Datensatz/Artikels Herkunft: [name] aus Artikelstamm (BLE-Online Shop)
MediumTraeger	medientraeger	String (255)	Ja	Medienträger des Datensatz/Artikels

				Herkunft: [medientraeger] aus Artikelstamm (BLE-Online Shop)
MediumTyp	medientype	String (255)	Ja	Medientyp des Datensatz/Artikels Herkunft: [medientyp] aus Artikelstamm (BLE-Online Shop)
Auflage	auflage	String (255)	Ja	Auflagenart des Datensatz/Artikels Herkunft: [auflage] aus Artikelstamm (BLE-Online Shop)
Referat	devision	String (255)	Ja	Ausgabewert des Referats / Referatsbezeichnung Herkunft: [devision] aus Mandant aus Fachbereich (BLE-Online Shop)

Tabelle 7: Einzelner Artikelstamm - Attributdarstellung

[AN 5] Es muss im BLE-Online Shop in der Artikeldetailansicht einen Button zum Übertragen der Artikelstammdaten geben.

Der Button wird unter folgenden Bedingungen angezeigt:

- Der Artikel ist vom Typ Einfacher Artikel oder Einfacher-Downloadartikel sein
- VDL muss in Konfiguration aktiviert sein
- Artikel muss bereits einmal gespeichert worden sein

[AN 6] Es müssen die folgenden Aufrufe implementiert werden:
POST [my.domain.local]/api/rest/products – Artikelstammdatenpflege

[AN 7] Der Request /api/rest/products zum Anlegen von Artikelstammdaten:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	{ "entity_id": int : { "entity_id": int, "client": string, "origin_sku": string, "name": string,	

		<pre> "medientraeger": string, "medientype": string, "auflage": string, "division": string } } </pre>	
--	--	---	--

Tabelle 8: Einzelner Artikelstamm - REST Request

[AN 8] Der Response aus [AN 7] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	<pre> { "entity_id": int : { "entity_id": int, "client": string, "origin_sku": string, "name": string, "medientraeger": string, "medientype": string, "auflage": string, "division": string } } </pre>	x

Tabelle 9: Einzelner Artikelstamm - REST Response bei erfolgreicher Artikelstammdatenabfrage

Beispiel: POST [my.domain.local]/api/rest/products

```

{
  „entity_id“: „101“,
  "client": „BOEL“,
  „origin_sku“: "1234ABC",
  "name": "Testartikel",
  "medientraeger": "download",
  "medientype": "typ",
  "auflage": "1",
  "division": "631"
}

```

[AN 9] Den Response aus [AN 8] muss der BLE-Online Shop dem Bearbeiter als verständlichen Text als Erfolgsmeldung anzeigen.

[AN 10] Der Response aus [AN 7] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
error	Array	Enthält code & message	x
code	Integer	Fehlercode	
message	String	Fehlermeldung	

Tabelle 10: Einzelner Artikelstamm - REST Response bei fehlerhafter Artikelstammdatenabfrage

Beispiel: {

```

  "error": [
    {
      "code": 400,
      "message": "Resource error."
    }
  ]
}
```

[AN 11] Den REST-Response aus [AN 10] muss der BLE-Online Shop dem Bearbeiter mit einem verständlichen Text als Fehlermeldung anzeigen.

Aktuell gibt es folgende Medienträger:

- Sonstiges
- CD-ROM
- DVD
- Abo
- nur als Download
- Ringordner
- Kassette
- Medienpaket
- Print

Aktuell gibt es folgende MedienTypen:

- (Lager) Material für Medienpaket
- Abreißblock
- Aktionsideen
- App
- Audio
- Aufkleber
- Aufkleber
- Ausstellung
- Ausstellung
- Ausstellungselement
- Beratungsmaterial
- Broschüre
- Computerlernprogr mm
- Computerprogramm
- Dias
- Display
- Flyer
- Folien
- Forum
- Fotokarten
- Fotos
- Fragebogen
- Heft
- Infoblatt
- Infostand

- Internet
- Internet Special
- Katalog
- Kompaktinfo
- Leittext
- Magazinbeitrag
- Magnete
- Materialkarten
- Mediensammlung
- Messen
- Nachschlagewerk
- Pocket
- Poster
- Postkarte
- Presseinfo
- Pressekonferenz
- Präsentation
- Schülerheft
- Seminar
- Sendefähiger Beitrag
- Sprachbox
- Special
- Spiel
- TV-Film/TV-Beitrag
- TV-Spot
- Tagung
- Taschencomputer
- Theaterproduktion
- Tischaufsteller
- Typ-Test
- Unterrichtsmaterial
- Versandkosten
- Video
- Wanderausstellung
- Web-Seminar
- Werbung
- Zeitschrift
- Zubehör

Umgang im Fehlerfall:

- Von Seiten des AN ist ein Ticket an die Queue „Shop-System – Eingang“ zur weiteren Abstimmung und Klärung mit dem AG zu erstellen.
- Prüfung durch AG
- AG kann bei notwendigem Bedarf entsprechende Logs beim Betreiber des BLE-Online Shops anfordern

5. Lagerbestand

5.1. Funktion: Lagerfreigabeauftrag

Beschreibung:	Die BLE übersendet mittels Lagerfreigabeauftrag über die Schnittstelle im Bedarfsfall Anweisungen im Umgang mit Artikelstammdaten zur Auftragsabwicklung an den Vertriebsdienstleister.
Weg:	BLE-Online Shop → Vertriebsdienstleister
Aktivität:	BLE-Online Shop sendet alle Daten (push)
Übertragungsart:	SOAP, Liste
Übertragungs- häufigkeit:	Nach aktiver Bestätigung/Massenverarbeitung
Prozess:	1. BLE-Online Shop (Medienservice) sendet Daten 2. Vertriebsdienstleister empfängt Daten 3. und führt Beauftragung an den betroffenen [ShopProduktIDs] aus
Konsequenz:	Lagerbestandsveränderung, Versandfreigabe

Tabelle 11: Lagerfreigabeauftrag - Übersicht

Feld	Feld	Typ	Pflicht	Erläuterung
ShopProduktID	entity_id	Int unsigned	Ja	Eindeutige Zuweisungsnummer des Datensatz/Artikels Herkunft: [entity_id] Product_ID aus Artikelstamm (BLE-Online Shop)
Buchung	posting	String (255)	Ja	Anweisung/Auftrag zur Verarbeitung beim Vertriebsdienstleister Herkunft: [posting] Auswahl aus vordefinierter Liste (BLE-Online Shop)
Bemerkungstext	notice	Text	Nein	Freitextfeld Herkunft: [notice] aus Benutzer-Eingabe (BLE-Online Shop)
Menge	qty	Int unsigned	Nein	Optionale Angabe einer ganzzahligen Menge für Stückzahl Herkunft: [qty] aus Lagerfreigabe (BLE-Online Shop)
Fixdatum	fix_date	String (255)	Nein	Optionales Ausführungsdatum der Lagerumbuchung

				Herkunft: [fix_date] aus Lagerfreigabe (BLE-Online Shop)
--	--	--	--	--

Tabelle 12: Lagerfreigabeauftrag - Attributdarstellung

[AN 12] Es muss im BLE-Online Shop unter Katalog einen neuen Menüpunkt „Vertriebswesen“ geben.

[AN 13] Innerhalb von [AN 12] muss es ein Grid zur Darstellung von Lagerfreigabeaufträgen geben.

- Artikel
- Buchung
- Menge
- Auszuführen am
- Auftragsdatum
- Bearbeiter

[AN 14] Innerhalb von [AN 12] muss es einen Button zum Anlegen neuer Lagerfreigabeaufträge geben.

[AN 15] Das Formular zum Anlegen neuer Lagerfreigabeaufträge muss Eingabefelder für zwei verschiedene Lagerfreigabeaufträge besitzen (alte / neue Auflage).

[AN 16] Das Formular muss eine oder mehrere Zeilen mit folgenden Datenfeldern enthalten:

- Auswahl der Auflage (Artikel)
- Auswahl der Buchung aus DropDown-Box (vordefinierte Liste)
- Bemerkungstext als Textarea
- Auszuführen am
- Menge als Textinput

[AN 17] Nach dem Speichern muss sich der Auftrag im Status „Angefragt“ befinden.

[AN 18] Ein Auftrag wird erst (automatisch) versendet, wenn er im Status „Akzeptiert“ ist.

[AN 19] Der Status „Akzeptiert“ darf nicht vom selben Mitarbeiter gesetzt werden, der den Auftrag erstellt hat.

[AN 20] Die Aufträge im Status „Akzeptiert“ müssen automatisch versendet werden sobald der Zeitpunkt „Auszuführen am“ erreicht ist.

[AN 21] Beim Klick auf eine Zeile im Grid aus [AN 13], öffnet sich das Formular aus [AN 16] im Readonly-Modus und enthält genau eine Zeile zu den Details des Auftrags.

[AN 22] Es müssen die folgenden Aufrufe implementiert werden:
POST [my.domain.local]/api/rest/stockorder – Lagerfreigabeauftrag

[AN 23] Der Request */api/rest/stockorder* zur Übermittlung eines Lagerfreigabeauftrags:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	{ "entity_id": int { "entity_id": int, "posting": string, "notice": text, "qty": int, "fix_date": string	

		}	
		}	

Tabelle 13: Lagerfreigabeauftrag - REST Request

Beispiel: POST [my.domain.local]/api/rest/stockorder

```
{
  "1": {
    "entity_id": "1",
    "posting": "lorem ipsum",
    "notice": "Testbuchung",
    "qty": "3",
    „fix_date“: "2025-12-31"
  },
  "2": {
    "entity_id": "2",
    "posting": "lorem ipsum",
    "notice": "Artikel nicht mehr lieferbar und Nachdruck nicht geplant",
    "qty": "5",
    „fix_date“: "2025-12-31"
  }
}
```

[AN 24] Der Response aus [AN 23] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
success	Array		
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x
error	Array	Enthält code & message	
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x

Tabelle 14: Lagerfreigabeauftrag - REST Response bei erfolgreichem Lagerfreigabeauftrag

Beispiel: POST [my.domain.local]/api/rest/stockorder

```
{
  "success": [
    {
```

```

        "code": "200",
        "message": "Stock order created",
        "item_id": "2",
      }
    ],
    "error": [
      {
        "code": 400,
        "message": "Resource error.",
        "item_id": "2",
      }
    ]
  ]
}

```

[AN 25] Den Response aus [AN 24] muss der BLE-Online Shop dem Bearbeiter mit einem verständlichen Text als Erfolgsmeldung anzeigen.

[AN 26] Der Response aus [AN 23] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
success			
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x
error	Array	Enthält code & message	
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x

Tabelle 15: Lagerfreigabeauftrag - REST Response bei fehlerhaftem Lagerfreigabeauftrag

Beispiel: {

```

  "error": [
    {
      "code": 400,
      "message": "Resource error.",
      "item_id": "2",
    }
  ]
}

```

```
]
}
```

[AN 27] Den REST-Response aus [AN 26] muss der BLE-Online Shop dem Bearbeiter mit einem verständlichen Text als Fehlermeldung anzeigen. Zusätzlich muss der Fehler in einem Log File abgespeichert werden.

5.2. Funktion: Gesamter Lagerbestand

Beschreibung:	Der Vertriebsdienstleister verwaltet die physikalischen Lagerbestände auf unterschiedlichen Lagerplätzen je Artikeldatensatz. Um eine tägliche Plausibilität der Lagerbestände beider Systeme gewährleisten zu können, gilt es, zweimal täglich einen gesamten Abgleich auf Veränderungen durchzuführen. Hierzu zieht sich der BLE-Online Shop die entsprechenden Daten vom Vertriebsdienstleister (pull), prüft diese auf Veränderung und verarbeitet die Veränderungen auf den unterschiedlichen Lagerplätzen.
Weg:	BLE-Online Shop ← Vertriebsdienstleister
Aktivität:	BLE-Online Shop zieht sich alle Daten (pull)
Übertragungsart:	REST, Liste
Übertragungshäufigkeit:	1x 05:00 Uhr, 1x 12:00 Uhr
Handeln:	1. BLE-Online Shop zieht Daten, 2. prüft auf Abweichungen (Plausibilität) 3. und verarbeitet Abweichungen anhand eindeutiger [ShopProduktID]
Konsequenz:	Update Artikeldatensatz/Lagerwesen

Tabelle 16: Gesamter Lagerbestand - Übersicht

Bezeichnung	Feld	Typ	Pflicht	Erläuterung
ShopProduktID	entity_id	Int unsigned	Ja	Eindeutige Zuweisungsnummer des Datensatz/Artikels Herkunft: [entity_id] Product_ID aus Artikelstamm (BLE-Online Shop)
Gewicht	weight	Int	Ja	Artikelgewicht pro Stück in Gramm
DatensatzAktivVDL	vdل_productactive	Int	Ja	Angabe ob Artikelstamm beim VDL „aktiv“ = 1 oder „inaktiv/H“ = 0 geführt wird
Menge Eingangslager	qty_staging	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager

				„Eingangslager“ = gesperrt für den Versand Herkunft: [qty_staging] aus Artikelstamm (BLE-Online Shop)
Stellplatzanzahl Eingangslager	size_staging	Int	Ja	Anzahl in Paletten-Plätzen des Datensatz/Artikels auf dem Lager „Eingangslager“ = gesperrt für den Versand Herkunft: [size_staging] aus Artikelstamm (BLE-Online Shop)
Aktualisierung Eingangslager	qty_staging_updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Eingangslager“ geändert wurde
Menge Hauptlager	qty_main	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Hauptlager“ = aktiv für Versand Herkunft: [qty_main] aus Artikelstamm (BLE-Online Shop)
Stellplatzanzahl Hauptlager	size_main	Int	Ja	Anzahl in Paletten-Plätzen des Datensatz/Artikels auf dem Lager „Hauptlager“ = aktiv für Versand Herkunft: [size_main] aus Artikelstamm (BLE-Online Shop)
Aktualisierung Hauptlager	qty_main_updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des

				„Hauptlagers“ geändert wurde
Menge Sperrlager	qty_locked	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Sperrlager“ = gesperrt für den Versand Herkunft: [qty_locked] aus Artikelstamm (BLE-Online Shop)
Stellplatzanzahl Sperrlager	size_locked	Int	Ja	Anzahl in Paletten-Plätzen des Datensatz/Artikels auf dem Lager „Sperrlager“ = gesperrt für den Versand Herkunft: [size_locked] aus Artikelstamm (BLE-Online Shop)
Aktualisierung Sperrlager	qty_locked_updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Sperrlager“ geändert wurde
Menge Makulierung Papier	qty_mac_paper	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Makulierungslager Papier“ = gesperrt für Versand Herkunft: [qty_mac_paper] aus Artikelstamm (BLE-Online Shop)
Stellplatzanzahl Makulierung Papier	size_mac_paper	Int	Ja	Anzahl in Paletten-Plätzen des Datensatz/Artikels auf dem Lager „Makulierungslager

				Papier“ = gesperrt für Versand Herkunft: [size_mac_paper] aus Artikelstamm (BLE- Online Shop)
Aktualisierung Makulierung Papier	qty_mac_paper _updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Makulierungslager Papier“ geändert wurde
Menge Makulierung Rest	qty_mac_rest	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Makulierungslager Reststoffe“ = gesperrt für Versand Herkunft: [qty_mac_rest] aus Artikelstamm (BLE- Online Shop)
Stellplatzanzahl Makulierung Rest	size_mac_rest	Int	Ja	Anzahl in Paletten- Plätzen des Datensatz/Artikels auf dem Lager „Makulierungslager Reststoffe“ = gesperrt für Versand Herkunft: [size_mac_rest] aus Artikelstamm (BLE- Online Shop)
Aktualisierung Makulierung Rest	qty_mac_rest _updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Makulierungslager Reststoffe“ geändert wurde

Tabelle 17: Gesamter Lagerbestand - Attributdarstellung

Der Bestand „0“ als Stückzahl gilt auch als zulässiger Wert.

[AN 28] Es müssen die folgenden REST-Aufrufe implementiert werden:
GET [my.domain.local]/api/rest/stockitems – Lagerbestandsabfrage

[AN 29] Der REST-Request */api/rest/stockitems* zur Abfrage von Lagerbestandsdaten, hat folgenden Aufbau:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
URL Parameter			
limit	int	Default: 10 Maximum: 100	
page	int	Seitenzahl, für die Ergebnisse zurückgegeben werden sollen	

Tabelle 18: Gesamter Lagerbestand - REST Request

[AN 30] Der REST-Response aus [AN 29] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	<pre>{ "entity_id": int : { "entity_id": int, "weigh": int, "vdl_productactive": int, "qty_staging": int, "size_staging": int, "qty_staging_updated_at": string, "qty_main": int, "size_main": int, "qty_main_updated_at": string, "qty_locked": int, "size_locked": int, "qty_locked_updated_at": string,</pre>	x

		<pre> “qty_mac_paper”: int, “size_mac_paper”: int, “qty_mac_paper_updated_at”: string, “qty_mac_rest”: int, “size_mac_rest”: int, “qty_mac_rest_updated_at”: string } } </pre>	
--	--	--	--

Tabelle 19: Gesamter Lagerbestand - REST Response bei erfolgreicher Lagerbestandsabfrage

Beispiel: [my.domain.local]/api/rest/stockitems?limit=100&page=2

```

{
  „101“: {
    “entity_id”: “101”,
    ”weighth”: “375”,
    “vdl_productactive”: “1”,
    “qty_staging”: “0”,
    “size_staging”: “0”,
    “qty_staging_updated_at”: “2023-12-24T18:21Z”,
    “qty_main”: “5400”,
    “size_main”: “9”,
    “qty_main_updated_at”: “2023-12-24T18:21Z”,
    “qty_locked”: “5”,
    “size_locked”: “1”,
    “qty_main_locked_at”: “2023-12-24T18:21Z”,
    “qty_mac_paper”: “0”,
    “size_mac_paper”: “0”,
    “qty_mac_paper_updated_at”: “2023-12-24T18:21Z”,
    “qty_mac_rest”: “0”,
    “size_mac_rest”: “0”,
    “qty_mac_rest_updated_at”: “2023-12-24T18:21Z”
  },
  „102“: {
    “entity_id”: “102”,
    ”weighth”: “1052”,
    “vdl_productactive”: “0”,
    “qty_staging”: “40”,
    “size_staging”: “3”,
    “qty_staging_updated_at”: “2023-12-24T18:21Z”,
    “qty_main”: “10”,
    “size_main”: “7”,
    “qty_main_updated_at”: “2023-12-24T18:21Z”,
    “qty_locked”: “0”,
    “size_locked”: “0”,
    “qty_main_locked_at”: “2023-12-24T18:21Z”,
    “qty_mac_paper”: “0”,
    “size_mac_paper”: “0”,

```

```

    "qty_mac_paper_updated_at": "2023-12-24T18:21Z",
    "qty_mac_rest": "0",
    "size_mac_rest": "0",
    "qty_mac_rest_updated_at": "2023-12-24T18:21Z"
  }
}

```

[AN 31] Der REST-Response aus [AN 29] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
error	Array	Enthält code & message	x
code	Integer	Fehlercode	
message	String	Fehlermeldung	
entity_id	Integer	Id des Produkts	x

Tabelle 20: Gesamter Lagerbestand - REST Response bei fehlerhafter Artikelstammdatenabfrage

Beispiel: {

```

  "error": [
    {
      "code": 400,
      "message": "Resource collection filtering error."
    }
  ]
}

```

[AN 32] Den REST-Response aus [AN 31] muss in einem Log File abgespeichert werden. Zusätzlich muss die Fehlermeldung per E-Mail an einem konfigurierbaren Mitarbeiter übermittelt werden.

5.3. Funktion: Einzelner Lagerbestand

Beschreibung:	Der Vertriebsdienstleister verwaltet die physikalischen Lagerbestände auf unterschiedlichen Lagerplätzen je Artikeldatensatz. Bei entstandenen Lagerbestandsveränderungen auf Grund von Zugängen oder Abgängen aller Lagerplätze werden diese höchstens alle 5 Minuten an den BLE-Online Shop übermittelt (Ereignistrigger mit Zeitbeschränkung).
Weg:	Vertriebsdienstleister → BLE-Online Shop
Aktivität:	Vertriebsdienstleister liefert Daten (push)
Übertragungsart:	REST, Liste
Übertragungshäufigkeit:	Ereignisbasierte Übersendung von Veränderungen maximal alle 5 Minuten
Handeln:	1. Vertriebsdienstleister sendet Daten 2. BLE-Online Shop nimmt Daten an 3. und verarbeitet diese anhand eindeutiger [ShopProduktID]
Konsequenz:	Update Artikeldatensatz/Lagerwesen

Tabelle 21: Einzelner Lagerbestand - Übersicht

Bezeichnung	Feld	Typ	Pflicht	Erläuterung
ShopProduktID	entity_id	Int unsigned	Ja	Eindeutige Zuweisungsnummer des Datensatz/Artikels Herkunft: [entity_id] Product_ID aus Artikelstamm (BLE-Online Shop)
Gewicht	weight	Int	Ja	Artikelgewicht pro Stück in Gramm
DatensatzAktivVDL	vdل_productactive	Int	Ja	Angabe ob Artikelstamm beim VDL „aktiv“ = 1 oder „inaktiv/H“ = 0 geführt wird
Menge Eingangslager	qty_staging	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Eingangslager“ = gesperrt für den

				Versand Herkunft: [qty_staging] aus Artikelstamm (BLE- Online Shop)
Stellplatzanzahl Eingangslager	size_staging	Int	Ja	Anzahl in Paletten- Plätzen des Datensatz/Artikels auf dem Lager „Eingangslager“ = gesperrt für den Versand Herkunft: [size_staging] aus Artikelstamm (BLE- Online Shop)
Aktualisierung Eingangslager	qty_staging_updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Eingangslager“ geändert wurde
Menge Hauptlager	qty_main	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Hauptlager“ = aktiv für Versand Herkunft: [qty_main] aus Artikelstamm (BLE- Online Shop)
Stellplatzanzahl Hauptlager	size_main	Int	Ja	Anzahl in Paletten- Plätzen des Datensatz/Artikels auf dem Lager „Hauptlager“ = aktiv für Versand Herkunft: [size_main] aus Artikelstamm (BLE- Online Shop)
Aktualisierung Hauptlager	qty_main_updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Hauptlagers“ geändert wurde

Menge Sperrlager	qty_locked	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Sperrlager“ = gesperrt für den Versand Herkunft: [qty_locked] aus Artikelstamm (BLE-Online Shop)
Stellplatzanzahl Sperrlager	size_locked	Int	Ja	Anzahl in Paletten-Plätzen des Datensatz/Artikels auf dem Lager „Sperrlager“ = gesperrt für den Versand Herkunft: [size_locked] aus Artikelstamm (BLE-Online Shop)
Aktualisierung Sperrlager	qty_locked_updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Sperrlager“ geändert wurde
Menge Makulierung Papier	qty_mac_paper	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Makulierungslager Papier“ = gesperrt für Versand Herkunft: [qty_mac_paper] aus Artikelstamm (BLE-Online Shop)
Stellplatzanzahl Makulierung Papier	size_mac_paper	Int	Ja	Anzahl in Paletten-Plätzen des Datensatz/Artikels auf dem Lager „Makulierungslager Papier“ = gesperrt für Versand

				Herkunft: [size_mac_paper] aus Artikelstamm (BLE- Online Shop)
Aktualisierung Makulierung Papier	qty_mac_paper _updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Makulierungslager Papier“ geändert wurde
Menge Makulierung Rest	qty_mac_rest	Int	Ja	Menge in Stück des Datensatz/Artikels auf dem Lager „Makulierungslager Reststoffe“ = gesperrt für Versand Herkunft: [qty_mac_rest] aus Artikelstamm (BLE- Online Shop)
Stellplatzanzahl Makulierung Rest	size_mac_rest	Int	Ja	Anzahl in Paletten- Plätzen des Datensatz/Artikels auf dem Lager „Makulierungslager Reststoffe“ = gesperrt für Versand Herkunft: [size_mac_rest] aus Artikelstamm (BLE- Online Shop)
Aktualisierung Makulierung Rest	qty_mac_rest _updated_at	string	Ja	Zeitstempel (UTC ISO 8601) wann die Lagermenge des „Makulierungslager Reststoffe“ geändert wurde

Tabelle 22: Einzelner Lagerbestand - Attributdarstellung

[AN 33] Es müssen die folgenden REST-Aufrufe implementiert werden:
PUT [my.domain.local]/api/rest/vdl_stockitems – Lagerbestandspflege

[AN 34] Der REST-Request /api/rest/vdl_stockitems zur Übermittlung von Lagerbestandsänderung(en):

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	<pre>{ "entity_id": int : { "entity_id": int, "weigh": int, "vdl_productactive": int, "qty_staging": int, "size_staging": int, "qty_staging_updated_at": string, "qty_main": int, "size_main": int, "qty_main_updated_at": string, "qty_locked": int, "size_locked": int, "qty_locked_updated_at": string, "qty_mac_paper": int, "size_mac_paper": int, "qty_mac_paper_updated_at": string, "qty_mac_rest": int, "size_mac_rest": int, "qty_mac_rest_updated_at": string } }</pre>	

Tabelle 23: Einzelner Lagerbestand - REST Request

Beispiel: PUT [my.domain.local]/api/rest/vdl_stockitems

```
{
  „101“: {
    "entity_id": "101",
    "weigh": "375",
    "vdl_productactive": "1",
```

```

"qty_staging": "0",
"size_staging": "0",
"qty_staging_updated_at": "2023-12-24T18:21Z",
"qty_main": "5400",
"size_main": "9",
"qty_main_updated_at": "2023-12-24T18:21Z",
"qty_locked": "5",
"size_locked": "1",
"qty_main_locked_at": "2023-12-24T18:21Z",
"qty_mac_paper": "0",
"size_mac_paper": "0",
"qty_mac_paper_updated_at": "2023-12-24T18:21Z",
"qty_mac_rest": "0",
"size_mac_rest": "0",
"qty_mac_rest_updated_at": "2023-12-24T18:21Z"
}

```

[AN 35] Der REST-Response aus [AN 34] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
success			
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x
error	Array	Enthält code & message	
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x

Tabelle 24: Einzelner Lagerbestand - REST Response bei erfolgreichem Lagerfreigabeauftrag

Beispiel:

```

{
  "success": [
    {
      "item_id": "101",
      "message": "Stock item processed",
      "code": 200
    },
    {
      "item_id": "102",
      "message": "min_qty: Please enter a valid number in \"qty_main\" field",

```

```

    "code":400
  }
]
}

```

[AN 36] Der REST-Response aus 34] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
success			
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x
error	Array	Enthält code & message	
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
item_id	Integer	ShopProduktID	x

Tabelle 25: Einzelner Lagerbestand - REST Response bei fehlerhaftem Lagerfreigabeauftrag

Beispiel:

```

{
  "error": [
    {
      "item_id":"101",
      "message":"min_qty: Please enter a valid number in \"qty_main\" field",
      "code":400
    },
    {
      "item_id":"102",
      "message":"min_qty: Please enter a valid number in \"qty_main\" field",
      "code":400
    }
  ]
}

```

[AN 37] Den REST-Response aus [AN 36] muss in einem Log File abgespeichert werden. Zusätzlich muss die Fehlermeldung per E-Mail an einem konfigurierbaren Mitarbeiter übermittelt werden.

6. Bestellungen

6.1. Funktion: Einzelne Bestellung

Beschreibung:	Der BLE-Online Shop nimmt alle Bestellungen des Kunden entgegen und liefert alle auslieferungsrelevanten Daten an den Vertriebsdienstleister zur weiteren Verarbeitung.
Weg:	BLE-Online Shop → Vertriebsdienstleister
Aktivität:	BLE-Online Shop liefert Daten (push)
Übertragungsart:	SOAP, Einzeln
Übertragungs- häufigkeit:	Unmittelbar
Handeln:	1. BLE-Online Shop (Medienservice) liefert Daten 2. Vertriebsdienstleister nimmt Daten an 3. und verarbeitet diese anhand eindeutiger [Bestellnummer]
Konsequenz:	Neuer Bestelldatensatz, Verarbeitung

Tabelle 26: Einzelne Bestellung - Übersicht

Bezeichnung	Feld	Typ	Pflicht	Erläuterung
BestellID	entity_id	Int unsigned	Ja	Eindeutige BestellID des Datensatz/Bestellung Herkunft: [entity_id] → rowid aus Bestellung (BLE-Online Shop)
Bestellnummer	order_increment_id	String (255)	Ja	Bestellnummer des Datensatz/Bestellung im BLE-Online Shop Herkunft: [order_increment_id] aus Bestellung (BLE-Online Shop)
Rechnungsbetrag Brutto EUR	total_invoiced	float	Ja	Rechnungsbetrag, im Zweifel auch 0,00 EUR oder 3,00 EUR/5,00 EUR/11,00EUR wenn

				nur Versandkosten angefallen sind. Hintergrund: [total_invoiced] aus Bestellung (BLE-Online Shop) zur Ausweisung auf Steuerunterlagen insbesondere in Drittländern und die Vorhaltefrist.
MwSt Betrag EUR	tax_invoiced	float	Ja	Mehrwertsteuerbetrag der auf den Rechnungsbetrag angefallen ist in EUR. Auch 0,00 wenn so ist. Hintergrund: [tax_invoiced] aus Bestellung (BLE-Online Shop) zur Ausweisung auf Steuerunterlagen insbesondere in Drittländern und die Vorhaltefrist.
Warenwert EUR	total_goodsvalue	float	Ja	Warenwert (Netto) in Summe über die gesamte Bestellung. Hintergrund: [total_goodsvalue] aus Bestellung (BLE-Online Shop) zur Ausweisung auf Steuerunterlagen insbesondere in Drittländern und die Vorhaltefrist.
Bestelldatum	created_at	date	Ja	Bestelldatum des Datensatz/Bestellung

				Herkunft: [created_at] Datum aus Bestellung (BLE-Online Shop)
Lieferanmerkung	vdl_note	Text	Nein	Optionale BLE- Bemerkung für Anlieferung Herkunft: [vdl_note] aus Bestellung (BLE-Online Shop)
Ankunftsdatum	vdl_delivery_date	date	Nein	Datumswert, wann die Lieferung angekommen sein muss (Fixdatum) Herkunft: [vdl_delivery_date] aus Bestellung (BLE-Online Shop)
Abholdatum	vdl_pickup_date	date	Nein	Datumswert, wann die Lieferung abgeholt werden muss (Fixdatum) Herkunft: [vdl_pickup_date] aus Bestellung (BLE-Online Shop)
Kunden- Bestellkennzeichen	ext_order_id	String (18)	Nein	Kundenbestellreferenz Herkunft: [ext_order_id] aus Bestellung (BLE- Online Shop)
Kundennummer	customer_increment_id	String (50)	Ja	Eindeutige Kunden_ID des Kunden im BLE- Online Shop Herkunft: [customer_increment_id] aus Kundenkonto (BLE- Online Shop)
Anrede	prefix	String (50)	Nein	Anrede des Kunden im BLE-Online Shop Herkunft: [prefix] aus Kundenkonto (BLE- Online Shop)

AkademischerTitel	academic_title	String (50)	Nein	Akademischer Titel des Kunden im BLE-Online Shop Herkunft: [academic_title] aus Kundenkonto (BLE-Online Shop)
Vorname	firstname	String (Summe 40*)	Ja	Vorname des Kunden der Lieferanschrift Herkunft: [firstname] aus Lieferadresse (BLE-Online Shop)
Nachname	lastname	String (Summe 40*)	Ja	Nachname des Kunden der Lieferanschrift Herkunft: [lastname] aus Lieferadresse (BLE-Online Shop)
Firma1	company	String (40)	Nein	Firma der Lieferanschrift Herkunft: [company] aus Lieferadresse (BLE-Online Shop)
Firma2	company2	String (40)	Nein	Firma2 der Lieferanschrift Herkunft: [company2] aus Lieferadresse (BLE-Online Shop)
Strasse_Hnr	street	String (40)	Ja	Straße und Hausnummer der Lieferanschrift Herkunft: [street] aus Lieferadresse (BLE-Online Shop)
Stadt	city	String (40)	Ja	Stadt der Lieferanschrift Herkunft: [city] aus Lieferadresse (BLE-Online Shop)
LandID	country_id	String (255)	Ja	Land der Lieferanschrift / ISO 3166 ALPHA-2

				Herkunft: [country_id] aus Lieferadresse (BLE- Online Shop)
Postleitzahl	postcode	String (10)	Ja	Postleitzahl der Lieferanschrift Herkunft: [postcode] aus Lieferadresse (BLE- Online Shop)
Telefonnummer	shipping_telephone	String (255)	Nein	Telefonnummer der Lieferanschrift Herkunft: [shipping_telephone] aus Lieferadresse (BLE- Online Shop)
e-Mail	shipping_email	String (255)	Nein	E-Mail der Lieferanschrift Herkunft: [shipping_email] aus Lieferadresse (BLE- Online Shop)
PostitionsID*	order_item_id	Int unsigned	Ja	Eindeutige Position des Datensatz/ Bestellposition Herkunft: [order_item_id] aus Bestellung (BLE-Online Shop)
ShopProduktID*	entity_id	Int unsigned	Ja	Eindeutige Zuweisungsnummer des Datensatz/Artikels Herkunft: Produkt ID [entity_id] aus Artikelstamm (BLE- Online Shop)
Artikelnummer*	origin_sku	String (255)	Nein	Artikelnummer des Datensatz/Artikels Herkunft: [origin_sku] Basisartikelnummer aus Artikelstamm (BLE- Online Shop)

Titel*	name	String (255)	Ja	Titelname des Datensatz/Artikels Herkunft: [name] aus Artikelstamm (BLE- Online Shop)
Auflage*	edition	String (255)	Ja	Auflagenart des Datensatz/Artikels Herkunft: [auflage] aus Artikelstamm (BLE- Online Shop)
ISBN_EAN*	gtin	String (255)	Nein	ISBN/EAN/GTIN des Datensatz/Artikels Herkunft: [gtin] aus Artikelstamm (BLE- Online Shop)
Bestellmenge*	qty_ordered	int	Ja	Bestellte Menge in Stück des Datensatz/Artikels Herkunft: [qty_ordered] aus Bestellung (BLE- Online Shop)
Label*	label_extern	String (255)	Ja	Mandantenbeschreibung des Datensatz/Artikels Herkunft: devision_id [external_label] aus Fachbereich (BLE- Online Shop)

Tabelle 27: Einzelne Bestellung - Attributdarstellung

*je Position/bestellter Artikel

[AN 38] Im BLE-Online Shop muss im Formular für den Fachbereich das Feld zur Angabe eines Labels hinzugefügt werden.

[AN 39] Es müssen die folgenden Aufrufe implementiert werden:
POST [my.domain.local]/api/rest/order – Bestellung

[AN 40] Der Request */api/rest/order* zur Übermittlung einer Bestellung:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	<pre>{ "entity_id": int, "order_increment_id": string, "total_invoiced": float, "tax_invoiced": float, "total_goodsvalue": float, "created_at": date (ISO8601), "vdl_note": string, "vdl_delivery_date": date (ISO8601), "vdl_pickup_date": date (ISO8601), "ext_order_id": string, "customer_increment_id": string, "prefix": string, "academic_title": string, "firstname": string, "lastname": string, "company": string, "company2": string, "street": string, "city": string, "country_id": string, "postcode": string, "shipping_telephone": string, "shipping_email": string, "order_items": ["order_item_id": int { "entity_id": int, "origin_sku": string, "name": string, "editon": string, "gtin": string, "qty_ordered": int, "label_extern": string }] }</pre>	

		]	
		}	
URL Parameter			

Tabelle 28: Einzelne Bestellung - REST Request

Beispiel: POST [my.domain.local]/api/rest/order

```
{
  "entity_id": "1",
  "order_increment_id": "1000000001",
  "total_invoiced": "50.50",
  "tax_invoiced": "08.06",
  "total_goodsvalue": "50.50",
  "created_at": "2023-05-25T15:52:01+0000",
  "vdl_note": "lorem ipsum",
  "vdl_delivery_date": "2023-05-25T15:52:01+0000",
  "vdl_pickup_date": "2023-05-25T15:52:01+0000",
  "ext_order_id": "123ABC",
  "customer_increment_id": "1",
  "prefix": "Herr",
  "academic_title": "Dr.",
  "firstname": "Wer",
  "lastname": "Immer",
  "company": "BLE",
  "company2": "",
  "street": "Deichmanns Aue 29",
  "city": "Bonn",
  "country_id": "Deutschland",
  "postcode": "53175",
  "shipping_telephone": "0228 8499 0",
  "shipping_email": "service@ble-medienservice.de",
  "order_items": [
    "1": {
      "entity_id": "1",
      "origin_sku": "TEST123",
      "name": "Testartikel",
      "editon": "1",
      "gtin": "12345678901234",
      "qty_ordered": "50",
      "label_extern": "lorem ipsum"
    },
    "2": {
      "entity_id": "5",
      "origin_sku": "TEST125",
      "name": "Testartikel5",
      "editon": "3",
      "gtin": "12345678901235",
      "qty_ordered": "20",
      "label_extern": "lorem ipsum"
    }
  ]
}
```

```
}
]
}
```

[AN 41] Der Response aus [AN 39] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
success	Array		
code	Integer	Fehlercode	x
message	String	Meldung	x

Tabelle 29: Einzelne Bestellung - REST Response bei erfolgreicher Bestellung

Beispiel: POST [my.domain.local]/api/rest/order

```
{
  "success": [
    {
      "code": "200",
      "message": "Order processed",
    }
  ]
}
```

[AN 42] Der Response aus [AN 39] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
error	Array	Enthält code & message	
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x

Tabelle 30: Einzelne Bestellung - REST Response bei fehlerhafter Bestellung

Beispiel:

```
{
  "error": [
    {
```

```
"code": 400,  
"message": "Resource error.",  
"item_id": "2",  
}  
]  
}
```

[AN 43] Den REST-Response aus [AN 39] muss in einem Log File abgespeichert werden. Zusätzlich muss die Fehlermeldung per E-Mail an einem konfigurierbaren Mitarbeiter übermittelt werden.

6.2. Funktion: Lieferstatus einzelner Bestellung inkl. Nachsendungen

Beschreibung:	Der Vertriebsdienstleister verarbeitet die über den BLE-Online Shop übermittelten Bestellungen und liefert Daten bzgl. Aller Lieferung (auch Nachsendungen zur Mangelbeseitigung) inkl. dem Dokument aller Lieferscheine an den BLE-Online Shop zurück.
Weg:	Vertriebsdienstleister → BLE-Online Shop
Aktivität:	Vertriebsdienstleister liefert Daten (push)
Übertragungsart:	REST, Einzeln
Übertragungs- häufigkeit:	Unmittelbar
Handeln:	1. Vertriebsdienstleister sendet Daten 2. BLE-Online Shop nimmt Daten an 3. und verarbeitet diese anhand eindeutiger [Bestellnummer]
Konsequenz:	Update Bestellungen

Tabelle 31: Lieferstatus Bestellung - Übersicht

Bezeichnung		Typ	Pflicht	Erläuterung
BestellID	entity_id	Int unsigned	Ja	Eindeutige BestellID des Datensatz/Bestellung Herkunft: [entity_id] aus Bestellung (BLE-Online Shop)
Sendungsnummer	track_number	String (255)	Nein	Eindeutige Sendungsnummer für die Nachverfolgen, wenn Brief dann leer
Transportdienstleister	carrier_code	String (32)	Ja	Code des verwendeten Transportdienstleister
URLSendung	tracking_url	String (255)	Nein	Sendungslink zur Verfolgung, wenn Brief dann leer
DokLieferschein	shipping_label	Text	Ja	Übersendung Dokument „Lieferschein“ (PDF); codiert in Base64
Lieferscheinnummer	increment_id	String (255)	Ja	Nummer des Lieferscheins
Lieferdatum	created_at	date	Ja	Datum der Lieferung

PostitionsID*	order_item_id	Int unsigned	Ja	Eindeutige Position des Datensatz/Bestellposition Herkunft: [order_item_id] aus Bestellung (BLE-Online Shop)
Liefermenge*	qty	int	Ja	Anzahl gelieferte Menge des Datensatz/Artikels

Tabelle 32: Lieferstatus Bestellung - Attributdarstellung

[AN 44] Es müssen die folgenden REST-Aufrufe implementiert werden:

POST [my.domain.local]/api/rest/vdl_shipping – Lieferstatus

[AN 45] Der REST-Request */api/rest/vdl_shipping* zur Übermittlung eines Lieferstatus:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Authorization	String	OAuth-Auth-Data	x
Content-Type	String	application/json; charset=utf-8	x
Body			
	JSON	<pre>{ "entity_id": int, "track_number": string, "carrier_code": string, "tracking_url": string, "shipping_label": string, "increment_id": string, "created_at": date (ISO8601), "order_items": ["order_item_id": int, "qty": int,] }</pre>	

Tabelle 33: Lieferstatus Bestellung - REST Request

Beispiel: POST [my.domain.local]/api/rest/shipping

```
{
  "entity_id": "1",
  "track_number": "string",
```

```

"carrier_code": "string",
"tracking_url": "string",
"shipping_label": "string",
"increment_id": "string",
"created_at": "2023-05-25T15:52:01+0000",
"order_items": [
  {
    "order_item_id": "1",
    "qty": "50"
  },
  {
    "order_item_id": "5",
    "qty": "20"
  }
]
}

```

[AN 46] Der REST-Response aus [AN 44] ist ein JSON Objekt und enthält bei Erfolg folgende Elemente:

Name	Typ	Beschreibung/Daten	Pflicht
Header			
Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
success	Array		
code	Integer	Fehlercode	x
message	String	Meldung	x

Tabelle 34: Lieferstatus Bestellung - REST Response bei erfolgreicher Lieferinfoübermittlung

Beispiel: POST [my.domain.local]/api/rest/shipping

```

{
  "success": [
    {
      "code": "200",
      "message": "Shipping processed",
    }
  ]
}

```

[AN 47] Der REST-Response aus [AN 44] ist ein JSON Objekt und enthält bei einem Fehler folgende Elemente:

Name	Typ	Beschreibung	Pflicht
Header			

Content-Type	String	application/json; charset=utf-8	x
Body (JSON)			
error	Array	Enthält code & message	
code	Integer	Fehlercode	x
message	String	Fehlermeldung	x
entity_id	Integer	Id der Bestellung	x
created_at	String	Lieferdatum wenn Format ungültig ist	

Tabelle 35: Lieferstatus Bestellung - REST Response bei fehlerhafter Lieferinfoübermittlung

Beispiel: {

```

    "error": [
        {
            "code": 400,
            "message": "Resource error."
        }
    ]
}
```

[AN 48] Den REST-Response aus [AN 46] muss in einem Log File abgespeichert werden. Zusätzlich muss die Fehlermeldung per E-Mail an einem konfigurierbaren Mitarbeiter übermittelt werden.

Umgang im Fehlerfall:

- Von Seiten AN ist ein Ticket an die Queue „Shop-System – Eingang“ zur weiteren Abstimmung und Klärung mit dem AG zu erstellen.
- Prüfung durch AG
- AG kann bei notwendigem Bedarf entsprechende Logs beim Betreiber des BLE-Online Shops anfordern

7. Abbildungsverzeichnis

Abbildung 1: Kommunikation zwischen BLE-Online Shop (Auftraggeber - AG) und Vertriebsdienstleister (Auftragnehmer - AN)	4
Abbildung 2: Erläuterungsbeispiel Relation ProductID	5

8. Tabellenverzeichnis

Tabelle 1: Gesamter Artikelstamm - Übersicht	7
Tabelle 2: Gesamter Artikelstamm - Attributdarstellung.....	8
Tabelle 3: Gesamter Artikelstamm - Rest-Request	8
Tabelle 4: Gesamter Artikelstamm - REST Response bei erfolgreicher Artikelstammanfrage.....	9
Tabelle 5: Gesamter Artikelstamm - REST Response bei fehlerhafter Artikelstammanfrage.....	10
Tabelle 6: Einzelner Artikelstamm - Übersicht.....	11
Tabelle 7: Einzelner Artikelstamm - Attributdarstellung	12
Tabelle 8: Einzelner Artikelstamm - REST Request.....	13
Tabelle 9: Einzelner Artikelstamm - REST Response bei erfolgreicher Artikelstammanfrage	13
Tabelle 10: Einzelner Artikelstamm - REST Response bei fehlerhafter Artikelstammanfrage....	14
Tabelle 11: Lagerfreigabeauftrag - Übersicht.....	16
Tabelle 12: Lagerfreigabeauftrag - Attributdarstellung.....	17
Tabelle 13: Lagerfreigabeauftrag - REST Request	19
Tabelle 14: Lagerfreigabeauftrag - REST Response bei erfolgreichem Lagerfreigabeauftrag.....	19
Tabelle 15: Lagerfreigabeauftrag - REST Response bei fehlerhaftem Lagerfreigabeauftrag.....	20
Tabelle 16: Gesamter Lagerbestand - Übersicht.....	22
Tabelle 17: Gesamter Lagerbestand - Attributdarstellung.....	25
Tabelle 18: Gesamter Lagerbestand - REST Request.....	26
Tabelle 19: Gesamter Lagerbestand - REST Response bei erfolgreicher Lagerbestandsanfrage.....	27
Tabelle 20: Gesamter Lagerbestand - REST Response bei fehlerhafter Artikelstammanfrage....	28
Tabelle 21: Einzelner Lagerbestand - Übersicht.....	29
Tabelle 22: Einzelner Lagerbestand - Attributdarstellung	32
Tabelle 23: Einzelner Lagerbestand - REST Request.....	33
Tabelle 24: Einzelner Lagerbestand - REST Response bei erfolgreichem Lagerfreigabeauftrag	34
Tabelle 25: Einzelner Lagerbestand - REST Response bei fehlerhaftem Lagerfreigabeauftrag.....	35
Tabelle 26: Einzelne Bestellung - Übersicht.....	36
Tabelle 27: Einzelne Bestellung - Attributdarstellung.....	41
Tabelle 28: Einzelne Bestellung - REST Request.....	43
Tabelle 29: Einzelne Bestellung - REST Response bei erfolgreicher Bestellung	44
Tabelle 30: Einzelne Bestellung - REST Response bei fehlerhafter Bestellung	44
Tabelle 31: Lieferstatus Bestellung - Übersicht.....	46
Tabelle 32: Lieferstatus Bestellung - Attributdarstellung	47
Tabelle 33: Lieferstatus Bestellung - REST Request.....	47
Tabelle 34: Lieferstatus Bestellung - REST Response bei erfolgreicher Lieferinfoübermittlung.....	48
Tabelle 35: Lieferstatus Bestellung - REST Response bei fehlerhafter Lieferinfoübermittlung	49