



EshopWedrop integration API



Date: 2021-10-28

Version: 2.0

Inhaltsverzeichnis

1. Ziele der Integration.....	3
2. Erste Schritte.....	3
3. Zugriff auf LIVE- und Sandbox-Umgebungen.....	3
4. Client-Autorisierung.....	4
5. Bestellung erstellen und Etikett generieren.....	4
6. Bestellung stornieren.....	14
7. Bestellung für den Versand validieren.....	15
8. Zielländerliste abrufen.....	16
9. Ursprungsliste abrufen.....	16
10. Liste verfügbarer Liefermethoden abrufen.....	17
11. Bestellstatus abrufen.....	18
12. Versandmanifest generieren.....	19
13. Manifest herunterladen.....	21
14. Sendungsverfolgung.....	22
15. Etikett abrufen.....	23
16. Zusätzliche Dienste abrufen.....	23
17. Click-&-Collect-Punkte abrufen.....	24
18. Transportcharge schließen.....	25
19. Transportcharge abrufen.....	26
20. Transportchargendokument herunterladen.....	27
21. Zusätzliche Informationen.....	27

Allgemeine Anforderungen und Informationen

1. Ziele der Integration

Die Ziele sind, automatisch:

- Kundenbestellungen und Pakete im EshopWedrop-System bzw. im EshopWedrop-Kundenkonto zu registrieren
- Versandetiketten aus dem EshopWedrop-System in das Kundensystem zu übertragen (auf Paketebene);
- die Sendungsverfolgung von Kundenbestellungen und die Aktualisierung ihrer Status zu unterstützen;
- am Ende jedes Tages einen Versandmanifest für die Pakete des Kunden zu erstellen.

Bitte wenden Sie sich an den technischen Support, wenn Sie während der Entwicklung der Integration Unterstützung benötigen. support@eshopwedrop.com

2. Erste Schritte

Für die Entwicklung Ihrer Integration benötigen Sie Folgendes:

- Das API-Dokument – es enthält detaillierte Informationen zu den angebotenen Diensten. Bitte wenden Sie sich an den technischen Support, wenn Sie die neueste Version dieses Dokuments benötigen. support@eshopwedrop.com
- EshopWedrop-Benutzername, Passwort und API-Key – dies sind eindeutige Kennungen für Ihre Anwendung. Um Benutzername, Passwort und API-Key zu erhalten, wenden Sie sich bitte an EshopWedrop. Nach Prüfung Ihrer Anfrage stellen wir Ihnen Ihre Zugangsdaten bereit und richten den Zugriff auf unsere Testumgebung ein. support@eshopwedrop.com
- **Zugriff auf die Sandbox-Testumgebung – unsere Testumgebung ist eine vollständige Kopie der Produktionsumgebung (LIVE). Nach Abschluss Ihrer Integration und Erfüllung bestimmter Anforderungen erhalten Sie Benutzername, Passwort und API-Key für die Produktionsumgebung.**

3. Zugriff auf LIVE- und Sandbox-Umgebungen

Die LIVE-Umgebung ist erreichbar: <https://eswdb2b.com/admin>

Die Sandbox-Umgebung ist erreichbar: <https://sandbox.eswdb2b.com/admin>

Datenformat: JSON

4. Client-Autorisierung

Für die Autorisierung wird die Methode „HTTP Basic Auth“ verwendet. Das bedeutet, dass Sie bei jeder Anfrage Benutzername und Passwort (im Base64-Format gehasht) angeben müssen. Zusätzlich muss ein eindeutiger, von EshopWedrop bereitgestellter API-Key zur zusätzlichen Sicherheit im Header übergeben werden.

Authentifizierungsbeispiel:

```
curl -X POST \
/courier_v2/order/process \
-H 'Authorization: Basic {username:password hashed in base64}' \
-H 'Content-Type: application/json' \
-H 'api-key: {API_KEY}' \
-H 'cache-control: no-cache'
```

*Hinweis 1: Kontozugangsdaten und API-Key werden in einer separaten E-Mail bereitgestellt.
Hinweis 2: Zugangsdaten und API-Key in den Sandbox- und LIVE-Umgebungen stimmen nicht überein. Für beide Umgebungen werden unterschiedliche Zugangsdaten und API-Keys bereitgestellt.*

5. Bestellung erstellen und Etikett generieren

Eine Bestellung muss zusammen mit Produktinformationen registriert werden. Die Produktinformationen können dem unten beschriebenen Standarddatensatz entsprechen:

Request-URL-Endpunkt: https://sandbox.eswdb2b.com/api/courier_v2/order/process

5.1. Primärdaten

Variable	Type	Max_length	Value	Required	Erläuterung
destinations_id	number	3	Dynamic - see section 8	Yes	ID des Ziels, an das die Bestellung geliefert wird. Zielländer werden vor der Integration vereinbart und im Kundenkonto eingerichtet.

origins_id	number	3	Dynamic - see section 9	Yes	ID des Ursprungslands, in dem Bestellungen abgeholt werden. Ursprungsländer werden vor der Integration vereinbart und im Kundenkonto eingerichtet.
pickup_modules_id	number	3	Dynamic - see section 10	Yes	ID des ausgewählten Kuriermoduls. Kuriermodule werden vor der Integration vereinbart und im Kundenkonto eingerichtet.

5.2. Bestellspezifische Daten

Variable	Type	Max_length	Value	Required	Erläuterung
order_id	text / number	20		Yes	Kundenbestell-ID, maximal 20 Zeichen; muss eindeutig sein.
weight	number	(6,2)		No	Positive Zahl in kg. Das endgültige Bestellgewicht wird aus den Paketen berechnet.
volume	number	(9,3)		No	Optionale Zahl. Der endgültige Wert wird aus den Paketabmessungen berechnet.
currency	text	3		No	Währung der angegebenen Preise. Wenn keine Währung angegeben wird, wird der Standardwert „EUR“ verwendet.
total_price	Number	(10,2)		No	Gesamtpreis der Bestellung. Wenn keine Währung angegeben wird, wird der Standardwert „EUR“ verwendet.

cod	number	(6,2)		No	Wenn Nachnahme (COD) gilt, Betrag eingeben.
comment_text	text	200		No	Kommentar zur gesamten Bestellung, der an den Kurier weitergegeben wird.
insurance	number	(10,2)		No	Wenn eine Versicherung gilt, Betrag eingeben.
company	boolean	1	0,1	No	Den Wert auf 1 setzen, wenn die Bestellung von einem Unternehmen aufgegeben wurde (zusätzliche Unternehmensdaten aus Abschnitt 6.7 sind erforderlich).

5.3. Wenn die Bestellung von einem Unternehmen aufgegeben wird

JSON-Datenschlüssel: 'company_data'

Hinweis: Nur erforderlich, wenn 'company' auf 1 gesetzt ist; andernfalls kann dieser Abschnitt ausgelassen werden.

Variable	Type	Max_length	Value	Required	Erläuterung
name	text	60		Yes	Name des Empfängers (Firmenname, Personennamen usw.)
company_code	number	16		Yes	Handelsregisternummer des Unternehmens
company_person	text	40		Yes	Name der Ansprechperson, die ein Kurier anrufen kann

5.4. Daten der abholenden Person

JSON-Datenschlüssel: 'pickup_person'

Variable	Type	Max_length	Value	Required	Erläuterung
name*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Vorname der Person, die die Bestellung erhält
surname*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Nachname der Person, die die Bestellung erhält
phone	text	Venipak – 30		Courier-	Telefonnummer der

		DPD – 30 In_Post - 9		specific*	Person, die die Bestellung erhält. Die Vorwahl ist erforderlich (LT: +370, LV: +371, EE: +372...)
email	text	Venipak – 64 DPD – 64 In_Post - 64		Courier-specific*	E-Mail-Adresse der Person, die die Sendungsverfolgungsinformationen erhält.

* Die Anforderungen und Längenbegrenzungen der Felder für die abholende Person werden durch pickup_modules_id bestimmt. Senden Sie name, surname, phone und email, um kurrierspezifische Validierungsfehler zu vermeiden.

5.5. Paketspezifische Daten (für jedes Paket)

JSON-Datenschlüssel: 'parcels'

Variable	Type	Max_length	Value	Required	Erläuterung
parcel_id	text / number	20		Yes	ID des Pakets aus dem Kundensystem. Muss für jedes Paket und jede Anfrage eindeutig sein.
weight	number	(6,2)		Yes	Gewicht des Pakets. Maximal 6-stelliger Wert mit zwei Dezimalstellen. Muss in kg angegeben werden.
volume_weight	number	(8,2)		No	Volumengewicht des Pakets. Maximal 8-stelliger Wert mit zwei Dezimalstellen. Muss in m3 angegeben werden.
length	number			Yes	Länge des Pakets. Muss in cm angegeben werden. Nur ganze Zahl.
width	number			Yes	Breite des Pakets. Muss in cm angegeben werden. Nur ganze Zahl.
height	number			Yes	Höhe des Pakets. Muss in cm angegeben werden. Nur ganze Zahl.
package_type	number	1	1 – box 2 – bag 3 – other	No	Optionale positive ganze Zahl; bei Weglassen, leerem Wert oder 0 ist der Standardwert 1. Bestehende Werte bleiben

			4 – envelope 5 – pallet		aus Kompatibilitätsgründen erhalten.
price	number			No	Optionaler Paketwert.
hazardous	number			No	Optionale Kennzeichnung für Gefahrgut.

5.6. Zolldetails

JSON-Datenschlüssel: 'CustomsDetails'

Variable	Type	Max_length	Value	Required	Erläuterung
Currency	text			Yes	Nicht leerer Code der deklarierten Währung.
FreightCosts	number		>= 0	No	Frachtbetrag. Freight wird ebenfalls akzeptiert.
InsuranceCosts	number		>= 0	No	Versicherungsbet rag. Insurance wird ebenfalls akzeptiert.
AdditionalCosts	number		>= 0	Yes	Sonstige Zollkosten; 0 senden, wenn keine anfallen.
Importer	object			Yes	Angaben zum Importeur.
Importer.CompanyName	text			Yes	Name des Importeurunter nehmens oder Empfängers.
Importer.Contact Name	text			No	Name der Kontaktperson des Importeurs.

Importer.Address 1	text			Yes	Primäre Adresszeile des Importeurs.
Importer.Address 2	text			No	Zusätzliche Adresszeile des Importeurs.
Importer.City	text			Yes	Stadt des Importeurs.
Importer.State	text			No	Bundesland oder Region des Importeurs.
Importer.Postcode	text			Yes	Postleitzahl des Importeurs.
Importer.Country Code	text			Yes	Ländercode des Importeurs.
Importer.Phone	text			No	Telefonnummer des Importeurs.
Importer.Email	text			No	E-Mail-Adresse des Importeurs.
Details	array		non-empty	Yes	Deklarierte Waren.
Details[].Description	text			Yes	Produktbeschreibung.
Details[].UnitCode	text			No	Einheitencode des Produkts.
Details[].HsCode	text			Yes	Waren-/HS-Code.
Details[].UnitValue	number		>= 0	Yes	Wert je Einheit.
Details[].Quantity	number		> 0	Yes	Anzahl der Einheiten.
Details[].Country Of Manufacture	text			Yes	Herstellungsland.

Details[].NetUnitWeight	number		>= 0	Yes	Nettogewicht je Einheit.
Details[].GrossUnitWeight	number		>= 0	Yes	Bruttogewicht je Einheit.

5.7. Adressformat für Lieferungen an die Haustür (an die Kundenadresse)

JSON-Datenschlüssel: 'shipping'

Hinweis: Ein shipping-Objekt ist bei der Courier-v2-Validierung für jede Bestellung erforderlich. Der ausgewählte Kurier bestimmt die Anforderungen auf Feldebene.

Variable	Type	Max_length	Value	Required	Erläuterung
ship_name*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Vorname der Person, an die die Bestellung versandt wird
ship_surname*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Nachname der Person, an die die Bestellung versandt wird
postcode	number	Venipak – 5 DPD – 5 In_Post - 5		Courier-specific*	Die Anzahl der Ziffern der Postleitzahl hängt vom Empfängerland ab: EE – 5 Ziffern LT – 5 Ziffern LV – 4 Ziffern PL – 5 Ziffern RO – 6 Ziffern
street**	text	Venipak – 50 DPD – 40 In_Post - 50		Courier-specific*	Straße
building**	text	Venipak – 30 DPD – 30 In_Post - 30		Courier-specific*	Hausnummer
flat**	text	Venipak – 50 DPD – 50 In_Post - 50		Courier-specific*	Wohnungsnummer
city	text	Venipak – 40		Courier-	Stadt

		DPD – 40 In_Post - 40		specific*	
phone	number	Venipak – 20 DPD – 20 In_Post - 9		Courier-specific*	Telefonnummer der Person, an die die Bestellung versandt wird. Die Vorwahl ist erforderlich (LT: +370, LV: +371, EE: +372...)

* Anforderungen und Längenbegrenzungen für shipping name sind kurrierspezifisch. Senden Sie beide Werte, wenn verfügbar.

** Anforderungen an shipping address und kombinierte Längenbegrenzungen sind kurrierspezifisch. Senden Sie street, building, flat, city, postcode und phone entsprechend dem ausgewählten Kurier.

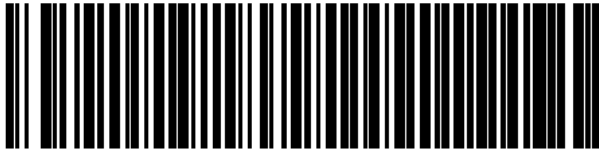
Hinweis: Wenn die Daten identisch sind, können ship_name, ship_surname und phone aus den Daten der abholenden Person übernommen werden.

5.8. Etikett generieren

Etiketten werden Base64-kodiert zurückgegeben. Die process-Antwort enthält ein Kurieretikett in file (ein Einzelwert oder ein Array, abhängig vom Kurier) sowie EshopWedrop-Tracking-/Etikettdaten in tracking_numbers. Drucken bzw. verwenden Sie die vom ausgewählten Kurier benötigten Etiketten.

Ein visuelles Beispiel für Kurieretiketten ist unten dargestellt:

5.10.1 Venipak-Etikett:

Sender: Eshopwedrop / Savanorių pr. 176, Vilnius LT-03154, VILNIUS, LT-03154;		 
Print date: 2019 03 20 Goods:		Directions: C.O.D.
Pack numeration: 1 \ 1		
Shipment No.: 024494743 Client shipment No.: 147854 Shipment Doc.No.: ESD0000000 Pack Doc.No.: P001017625		
Receiver: TESTING testing testing Gedimino pr. 125 - 25, Vilnius, VILNIUS, LT-19834 +44 7911 123456		
 V 0 9 1 8 6 E 0 1 3 0 9 7 3		
Route: 		

5.10.2. DPD-Etikett:

5.10.3. InPost-Etikett:

5.9. Beispiel einer POST-Anfrage:

Example {json_data} data with all available fields:

```

{
  "Consolidations":{
    "destinations_id":1,
    "origins_id":14,
    "pickup_modules_id":2,
    "order_id":"011111111",
    "weight":18,
    "currency":"EUR",
    "comment_text":"comment",
    "company":0,
    "company_data":{
      "company_code":"test",
      "company_person":"test"
    },
    "pickup_person":{
      "name":"test",
      "surname":"test",
      "phone":"+37067000000",
      "email":"test@email.lt"
    },
    "parcels":[
      {
        "parcel_id":"123456789",
        "package_type":"1",
        "weight":18,
        "length":31,
        "width":24,
        "height":19,
        "price":26
      }
    ],
    "CustomsDetails":{
      "Currency":"EUR",
      "FreightCosts":0,
      "InsuranceCosts":0,
      "AdditionalCosts":0,
      "Importer":{
        "CompanyName":"test",
        "Address1":"test street 1",
        "City":"Vilnius",
        "Postcode":"12345",
        "CountryCode":"LT"
      },
      "Details":[
        {
          "Description":"blue socks",
          "HsCode":"6115950000",
          "UnitValue":32,
          "Quantity":1,
          "CountryOfManufacture":"LT",
          "NetUnitWeight":0.1,
          "GrossUnitWeight":0.1
        }
      ]
    }
  }
}

```

```

    ],
    "shipping":{
      "ship_name":"test",
      "ship_surname":"test",
      "postcode":"12345",
      "street":"Gelezinio Vilko",
      "building":"18A",
      "flat":"",
      "city":"Vilnius",
      "phone":"+37067000000"
    }
  }
}

```

5.10. Beispiel der Antwort:

```

{
  "status": true,
  "order": "123456789",
  "tracking_numbers": {
    "PB000095034": {
      "courier_tracking_number": "05809023196443",
      "label": "XlhBXkZPMCwwXkdGQSwxODcwMDA....."
    }
  },
  "file": "XlhBXkZPMCwwXkdGQSwxMTc4MTAsM....."
}

```

5.11. Fehlermeldung:

```

{
  "status": false,
  "message": "Request data empty"
}

```

Hinweis: Tritt bei der Zollanmeldung ein Fehler auf, wird die gesamte Bestellung (alle Pakete) storniert.

6. Bestellung stornieren

Mit dieser Methode kann die in der Anfrage registrierte Bestellung storniert werden.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/cancel/{order_id}

Methode: GET

Daten: order_id – eindeutige interne Nummer, die der Kunde bei der Registrierung einer Bestellung über die API bereitstellt.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "message":"Order cancelled
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

7. Bestellung für den Versand validieren

Diese Methode validiert die Bestelldaten der Anfrage und gibt Erfolg zurück, wenn die Bestellung alle Einschränkungen erfüllt.

Request-URL:https://sandbox.eswdb2b.com/api/courier_v2/order/validate

Methode: POST

Daten: Die Validierungsmethode verwendet dieselben Anfragedaten wie die Bestellregistrierungsmethode (Abschnitt 5).

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "message":"Validation passed"
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
```

```
"message": {error_message}
}
```

8. Zielländerliste abrufen

Diese Methode gibt alle für den angeforderten API-Benutzer verfügbaren Ziele zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/get_destinations/{origins_id}

Methode: GET

Daten: origins_id – die Ursprungs-ID muss angegeben werden, um die verfügbare Zielliste abzurufen.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status": true,
  "message": "Success",
  "destinations": [
    {"destinations_id": "1", "name": "Lithuania DW", "country": "Lithuania", "country_code": "LT"},
    {"destinations_id": "2", "name": "Latvia DW", "country": "Latvia", "country_code": "LV"},
    {"destinations_id": "3", "name": "Estonia DW", "country": "Estonia", "country_code": "EE"}
  ]
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status": false,
  "message": {error_message}
}
```

9. Ursprungsliste abrufen

Diese Methode gibt alle für den angeforderten API-Benutzer verfügbaren Ursprungsländer zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/get_origins

Methode: GET

Daten: -

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "message":"Success",
  "origins":[
    {"origins_id":"2","name":"Italy OW"},
    {"origins_id":"3","name":"Poland OW"},
    {"origins_id":"6","name":"Lithuania OW"}
  ]
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

10. Liste verfügbarer Liefermethoden abrufen

Diese Methode gibt alle verfügbaren Last-Mile-Kuriere und deren Liefermethoden für das angeforderte Ziel zurück.

Request-URL:https://sandbox.eswdb2b.com/api/courier_v2/order/get_delivery_methods/{destinations_id}

Methode: GET

Daten: destinations_id – Ziel, für das die Liefermethoden zurückgegeben werden sollen.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "message":"Success",
  "pickups":{
    "1":{"pickup_modules_id":"1","name":"Flagship"},
    "2":{"pickup_modules_id":"2","name":"Door Venipak"},
    "3":{"pickup_modules_id":"3","name":"Click & Collect Venipak"}
  }
}
```

```
}}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

11. Bestellstatus abrufen

Diese Methode gibt die für das angeforderte Paket erfassten Statusverlaufereignisse zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/get_status/{parcel_id}

Methode: GET

Daten: parcel_id – ID des Pakets aus dem Kundensystem (Abschnitt 5.8).

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "list":[
    {
      "text":"{status_event}",
      "date":"{date}",
      "date_utc":"{date_utc}",
      "date_iso":"{date_iso}",
      "timezone":"{timezone}"
    }
  ]
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

Historische Statusbezeichnungen werden dynamisch zurückgegeben. Die folgende Liste bleibt als Referenz für ältere Daten erhalten und enthält möglicherweise nicht jedes Ereignis.

Status	Erläuterung
Ready for dispatch	<i>Bestellung wurde vom Kunden über API registriert</i>
Incoming to origin warehouse	<i>Bestellung ist auf dem Weg zum Ursprungsdepot (OW)</i>
In origin warehouse	<i>Bestellung ist im Ursprungsdepot (OW) angekommen</i>
Prepared for loading	<i>Bestellung ist für die Verladung auf den Lkw im OW vorbereitet</i>
In transit	<i>Bestellung hat das Ursprungsdepot verlassen</i>
Arrived to destination	<i>Bestellung wurde im Zieldepot (DW) gescannt</i>
Sent through courier	<i>Bestellung wurde vom Last-Mile-Kurier aus dem DW abgeholt</i>
Waiting for pick-up	<i>Bestellung wartet auf die Abholung durch den Kunden im Lager</i>
Completed	<i>Bestellung wurde an den Kunden geliefert</i>
Lost	<i>Bestellung ging verloren</i>
Cancelled	<i>Bestellung wurde storniert</i>

12. Versandmanifest generieren

Diese Methode generiert anhand der in der Anfrage angegebenen Datumsintervalle ein Manifest für alle über die API registrierten Bestellungen.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/manifests/generate

Methode: POST

Daten:

destinations_id - Ziel-Landes-ID des Manifests;

origins_id – Ursprungs-Landes-ID des Manifests;

date_from(date) – Bestellungen, die ab diesem Datum registriert wurden, werden in das Manifest aufgenommen;

date_to(date) – Bestellungen, die bis zu diesem Datum registriert wurden, werden in das Manifest aufgenommen;

sender_address(varchar(255)) – Absenderadresse, an der die Bestellungen abgeholt werden.

Anfragebeispiel:

```
{
  "destinations_id": [1],
  "origins_id": 2,
  "date_from": "2020-09-22",
  "date_to": "2020-09-24",
  "sender_address": "test address"
}
```

Wenn das Manifest für Pakete mehrerer Ziele generiert wird, geben Sie bitte alle Ziel-IDs an:

```
{
  "destinations_id": [1,2,3],
  "origins_id": 13,
  "date_from": "2019-01-01",
  "date_to": "2019-01-04",
  "sender_address": "test address"
}
```

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "barcode":"MH0000079702",
  "file":" {base_64_encoded_file}
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

Beispiel des PDF-Manifests:



Sender
test address

Receiver
EshopWedrop Baltics, Savanorių pr. 176, Vilnius LT-03154, Lithuania

Destination		
Vilnius LT	Manifest ID: MR0000079703 Date added: 2019-11-04 11:38	 MR0000079703

KPI	Value (%)
-----	-----------

Information	Value	Comment
Gross weight:	0.00 Kg	
Gross volume:	0.001 m ³	
Reference number:		
Trailer number:		
Truck number:		
Number of pallets:	1	

Total	Percentage
Actual weight	86.00 Kg
Chargeable weight	86.00 Kg
Volume	0.017 m ³
Parcels	4

Pallet ID: --

No.	Parcel id	Pallet ID	Country destination	Type	Dimensions			Actual weight	Volume
					Length	Width	Height		
1	P001156421		LT		12.00 cm	12.00 cm	15.00 cm	18.00 Kg	0.002 m ³
2	P001156422		LT		25.00 cm	16.00 cm	16.00 cm	25.00 Kg	0.006 m ³
3	P001156423		LT		12.00 cm	12.00 cm	15.00 cm	18.00 Kg	0.002 m ³
4	P001156424		LT		25.00 cm	16.00 cm	16.00 cm	25.00 Kg	0.006 m ³
Number of parcels: 4								Total:	86.00 Kg
									0.017 m ³

13. Manifest herunterladen

Diese Methode gibt die Base64-kodierte Datei eines zuvor generierten Manifests zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/manifests/download/{barcode}

Methode: GET

Daten: barcode – Manifest-Barcode aus der Generierungsantwort.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "file":{"base_64_encoded_file}
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

14. Sendungsverfolgung

Track-&-Trace-Links für Kunden sind unten aufgeführt. Diese Links können an Käufer weitergegeben werden, damit sie den Status ihrer Sendung selbst prüfen können.

Litauen:

<https://eshopwedrop.lt/delivery-options-lt#block-track>

Lettland:

<https://www.eshopwedrop.lv/delivery-options-lv#block-track>

Estland:

<https://www.eshopwedrop.ee/delivery-options-ee#block-track>

Polen:

<https://www.eshopwedrop.pl/delivery-options-ee#block-track>

Hinweis 1: Wenn der Kunde seinen Käufern einen Tracking-Link bereitstellen möchte, kann der Tracking-Link mit dem Tracking-Code zusammengeführt werden, sodass der Code nicht manuell kopiert werden muss. Beispiel eines zusammengeführten Links:

<https://eshopwedrop.lt/delivery-options-lt?parcel=1029032#block-track>

Erläuterung:

https://eshopwedrop.lt/delivery-options-lt - Link zu einer lokalen Website

?parcel=1029032 - Tracking-Code des Pakets

#block-track - Positionierung für automatisches Herunterscrollen

15. Etikett abrufen

Diese Methode gibt die Base64-kodierte Datei eines zuvor generierten Etiketts zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/label/{tracking_number}/{file_type}

Methode: GET

Daten: tracking_number – Tracking-Nummer der Bestellung oder des Pakets; file_type – pdf oder zpl.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status":true,
  "label":"{base_64_encoded_file}"
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status":false,
  "message": {error_message}
}
```

16. Zusätzliche Dienste abrufen

Diese Methode gibt die Zusatzdienstkonfiguration für das ausgewählte Ziel und die Liefermethode zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/get_additional_services/{destinations_id}/{pickup_modules_id}

Methode: GET

Daten: destinations_id – Ziel-ID aus „Zielländerliste abrufen“; pickup_modules_id – Liefermethoden-ID aus „Liste verfügbarer Liefermethoden abrufen“.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status": true,
  "message": "Success",
  "services": {
    "active": "1",
    "has_cod": "0",
    "ins_applied": "0",
    "has_pod": "0",
    "next_work_day_delivery": "0",
    "same_day_delivery": "0",
    "saturday_delivery": "0"
  }
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status": false,
  "message": "{error_message}"
}
```

17. Click-&-Collect-Punkte abrufen

Diese Methode gibt verfügbare Click-&-Collect-Abholpunkte für das ausgewählte Ziel und die Liefermethode zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/order/get_click_collect/{destinations_id}/{pickup_modules_id}

Methode: GET

Daten: destinations_id – Ziel-ID aus „Zielländerliste abrufen“; pickup_modules_id – Liefermethoden-ID aus „Liste verfügbarer Liefermethoden abrufen“. Die Felder der Abholpunkte sind kurrierspezifisch.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status": true,
  "click_collect": [
    {
      "{courier_pickup_point_field}": "{courier_pickup_point_value}"
    }
  ]
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status": false,
  "message": "{error_message}"
}
```

```
}
```

18. Transportcharge schließen

Diese Methode schließt eine Transportcharge, erstellt das Transportmanifest und gibt die generierten Dokumente zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/manifests/close_batch

Methode: POST

Daten:

Variable	Type	Max_length	Value	Required	Erläuterung
batch_reference	text	191	BATCH-20260715-001	Yes	Eindeutige Referenz für diese Transportcharge. Das Wiederholen derselben Anfrage ist idempotent; dieselbe Referenz mit anderen Daten führt zu einem Fehler.
cmr_truck_plate_number	text	64	B123ABC	Yes	Lkw-Kennzeichen/CMR-Kennzeichen.
origins_id	number		14	Yes	Von Get origins list zurückgegebene Ursprungs-ID. Jede Bestellung in order_ids muss zu diesem Ursprung gehören.
sender_address	text		Warehouse street 1, Vilnius	Yes	Absenderadresse, die auf den erzeugten Transportdokumenten gedruckt wird.
closed_at	datetime		2026-07-15 14:00:00	No	Zeitpunkt des Chargenabschlusses in der Zeitzone des Ursprungs, Format YYYY-MM-DD HH:mm:ss. Wenn nicht angegeben, wird die aktuelle Serverzeit verwendet.
order_ids	array of text / number		["01111111", "011111112"]	Yes	Externe Bestell-IDs. Bestellungen müssen zum authentifizierten Kunden gehören, versandbereit sein, noch keinem Manifest zugeordnet sein und dürfen nicht bereits zu einer Charge gehören.

Beispiel-Anfragedaten:

```
{
  "batch_reference": "BATCH-20260715-001",
  "cmr_truck_plate_number": "B123ABC",
  "origins_id": 14,
  "sender_address": "Warehouse street 1, Vilnius",
  "closed_at": "2026-07-15 14:00:00",
  "order_ids": ["011111111", "011111112"]
}
```

Beispiel für eine erfolgreiche Antwort:

```
{
  "status": true,
  "batch_reference": "BATCH-20260715-001",
  "cmr_truck_plate_number": "B123ABC",
  "closed_at": "2026-07-15 11:00:00",
  "orders_count": 2,
  "order_ids": ["011111111", "011111112"],
  "manifest_barcode": "MH0000079702",
  "documents": {
    "transport_manifest": {
      "file_name": "transport_manifest.pdf",
      "mime_type": "application/pdf",
      "content_base64": "{base_64_encoded_file}"
    }
  }
}
```

Beispiel für eine nicht erfolgreiche Antwort:

```
{
  "status": false,
  "message": "{error_message}"
}
```

19. Transportcharge abrufen

Diese Methode gibt Chargenmetadaten und Metadaten generierter Dokumente zurück. Sie enthält keinen Dokumentinhalt.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/manifests/batch/{batch_reference}

Methode: GET

Daten: batch_reference – beim Schließen der Charge angegebene Transportchargenreferenz.

Beispiel für eine erfolgreiche Antwort:

```
{
  "status": true,
  "batch_reference": "BATCH-20260715-001",
```

```

"cmr_truck_plate_number": "B123ABC",
"closed_at": "2026-07-15 11:00:00",
"orders_count": 2,
"order_ids": ["011111111", "011111112"],
"manifest_barcode": "MH0000079702",
"documents": {
  "transport_manifest": {
    "file_name": "transport_manifest.pdf",
    "mime_type": "application/pdf"
  }
}
}

```

Beispiel für eine nicht erfolgreiche Antwort:

```

{
  "status": false,
  "message": "{error_message}"
}

```

20. Transportchargendokument herunterladen

Diese Methode gibt ein generiertes Transportchargendokument als Base64-Inhalt zurück.

Request-URL: https://sandbox.eswdb2b.com/api/courier_v2/manifests/batch/{batch_reference}/document/{document_type}

Methode: GET

Daten: batch_reference – Transportchargenreferenz; document_type – commercial_invoice, packing_list, import_manifest, export_manifest, transport_manifest oder einer dieser Werte mit einem Ländersuffix (z. B. commercial_invoice_GB).

Beispiel für eine erfolgreiche Antwort:

```

{
  "status": true,
  "batch_reference": "BATCH-20260715-001",
  "manifest_barcode": "MH0000079702",
  "document": {
    "document_type": "transport_manifest",
    "file_name": "transport_manifest.pdf",
    "mime_type": "application/pdf",
    "content_base64": "{base_64_encoded_file}"
  }
}

```

Beispiel für eine nicht erfolgreiche Antwort:

```

{

```

```
"status": false,  
"message": "{error_message}"  
}
```

21. Zusätzliche Informationen

8.4 Bei Nutzung von C.O.D. (Nachnahme) müssen alle Beträge in die Währung des Kundenlandes umgerechnet werden. Diese Beträge werden von den Kurieren bei den Kunden eingezogen.

8.5 Tritt ein Fehler auf, gibt die Antwort status FALSE und eine Erläuterung der Fehlermeldung zurück.

8.6 Unser Webservice basiert auf einer REST-Architektur; das HTTP-Protokoll wird zum Austausch von Daten zwischen Anwendungen und Systemen verwendet. Alle ausgetauschten Daten müssen als JSON formatiert sein.

8.7 Zeichenkodierung: UTF-8

8.8 Bei optionalen Daten können Sie den Wert 0 (FALSE) senden oder den Schlüssel vollständig aus der Anfrage entfernen.

8.9 IT-Kontakte für Integrationszwecke und Integrationssupport:

Email: Support@eshopwedrop.com	

