



EshopWedrop integration API



Date: 2021-10-28

Version: 2.0

Table of content

1. The goals of integration.....	3
2. Getting started.....	3
3. Access to LIVE and Sandbox environments.....	3
4. Client authorization.....	4
5. Create an order and generate label.....	4
6. Cancel an order.....	14
7. Validate order for shipment.....	15
8. Get Destinations list.....	16
9. Get Origins list.....	16
10. Get list of available delivery methods.....	17
11. Get order status update.....	18
12. Generate shipment manifest.....	19
13. Download manifest.....	21
14. Order tracking.....	22
15. Get label.....	23
16. Get additional services.....	23
17. Get click & collect points.....	24
18. Close transport batch.....	25
19. Get transport batch.....	26
20. Download transport batch document.....	27
21. Additional information.....	27

General requirements and information

1. The goals of integration

The goals are to automatically:

- register client's orders and parcels in the EshopWedrop system / in EshopWedrop's client account
- transfer shipping labels from EshopWedrop system to client's system (at parcels level);
- support tracking of client's orders and update their statuses;
- generate shipment manifest for client's parcels shipped at the end of every day.

Please contact tech support at support@eshopwedrop.com, if you require assistance during development of integration.

2. Getting started

To begin developing your integration you will need the following:

- **The API document** – this will provide detailed information on the various services offered. Please contact tech support at support@eshopwedrop.com, if you require the latest version of this document.
- **Eshopwedrop Username, Password and API Key** – these are unique identifiers that represent your application. To get Username, Password and API Key please contact EshopWedrop at support@eshopwedrop.com. Once we have received and reviewed your request, we will provide you with your own details and work with you to set up your access to our test environment.
- **Access to Sandbox test environment** – our test environment is a complete replica of our Production (live) environment. Once you have completed your integration and meet certain requirements, you will be granted Username, Password and API Key for production environment.

3. Access to LIVE and Sandbox environments

Live environment can be accessed: <https://eswdb2b.com/admin>

Sandbox environment can be accessed: <https://sandbox.eswdb2b.com/admin>

Data format: JSON

4. Client authorization

For authorization, “HTTP Basic Auth” method is used, meaning you must provide the **Username** and **Password** (hashed in base64 format) for every request that is being made. In addition, a unique **API Key**, provided by EshopWedrop, should be passed in the header section for additional security.

Authentication example:

```
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'
```

Note 1: Account credentials and API Key are provided in a separate email.

Note 2: Credentials and API Key in Sandbox and LIVE environments do not match. Different credentials and API Keys will be provided for both environments.

5. Create an order and generate label

Order must be registered together with product information. Product information can be a default set of data as per instructions below:

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

5.1. Primary data

Variable	Type	Max_length	Value	Required	Explanation
destinations_id	number	3	Dynamic - see section 8	Yes	ID of the destination where the order will be delivered. Destination countries will be agreed prior to integration and set in client's account.
origins_id	number	3	Dynamic - see section 9	Yes	ID of the origin country where orders will be collected. Origin countries will

					be agreed prior to integration and set in client's account.
pickup_modules_id	number	3	Dynamic - see section 10	Yes	ID of the courier module selected. Courier modules will be agreed prior to integration and set in client's account.

5.2. Order specific data

Variable	Type	Max_length	Value	Required	Explanation
order_id	text / number	20		Yes	Client order ID, up to 20 characters; must be unique.
weight	number	(6,2)		No	Positive number, in kg. The final order weight is calculated from the parcels.
volume	number	(9,3)		No	Optional number. The final value is calculated from parcel dimensions.
currency	text	3		No	Currency of the provided prices. If no currency is provided, default value is selected: "EUR".
total_price	Number	(10,2)		No	Total price of the order. If no currency is provided, default value is selected: "EUR".
cod	number	(6,2)		No	If COD (cash on delivery) is applied, enter amount.
comment_text	text	200		No	Comment for the whole order that will be passed to courier.
insurance	number	(10,2)		No	If insurance is applied, enter amount.

company	boolean	1	0,1	No	Set the value to 1 if the order was made by company (additional company data will be required from section 6.7)
----------------	---------	---	-----	----	---

5.3. If order is made by company

JSON data-key: 'company_data'

Note: Only required if 'company' is set to 1, otherwise you can skip this.

Variable	Type	Max_length	Value	Required	Explanation
name	text	60		Yes	Consignee name (company name, person name etc)
company_code	number	16		Yes	Company registration code
company_person	text	40		Yes	Contact person name to whom a courier can call

5.4. Pick up person data

JSON data-key: 'pickup_person'

Variable	Type	Max_length	Value	Required	Explanation
name*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Name of the person who will receive the order
surname*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Surname of the person who will receive the order
phone	text	Venipak – 30 DPD – 30 In_Post - 9		Courier-specific*	Phone of the person who will receive the order. Prefix is mandatory (LT: +370, LV: +371, EE: +372...)
email	text	Venipak – 64 DPD – 64 In_Post - 64		Courier-specific*	Email of the person who will receive the tracking information.

* Pickup-person field requirements and length limits are selected by pickup_modules_id. Send name, surname, phone and email to avoid courier-specific validation errors.

5.5. Parcel specific data (for each parcel)

JSON data-key: 'parcels'

Variable	Type	Max_length	Value	Required	Explanation
parcel_id	text / number	20		Yes	ID of the parcel from the client's system. Must be unique for every parcel and every request.
weight	number	(6,2)		Yes	Weight of the parcel. Maximum 6 digit value with two decimals. Must be in kg.
volume_weight	number	(8,2)		No	Volume weight of the parcel. Maximum 8 digit value with two decimals. Must be in m ³ .
length	number			Yes	Length of the parcel. Must be in cm. Only whole number.
width	number			Yes	Width of the parcel. Must be in cm. Only whole number.
height	number			Yes	Height of the parcel. Must be in cm. Only whole number.
package_type	number	1	1 – box 2 – bag 3 – other 4 – envelope 5 – pallet	No	Optional positive integer; omitted, empty or 0 defaults to 1. Existing values are retained for compatibility.
price	number			No	Optional parcel value.
hazardous	number			No	Optional hazardous-goods flag.

5.6. Customs details

JSON data-key: 'CustomsDetails'

Variable	Type	Max_length	Value	Required	Explanation
Currency	text			Yes	Non-empty

					declared-currency code.
FreightCosts	number		>= 0	No	Freight amount. Freight is also accepted.
InsuranceCosts	number		>= 0	No	Insurance amount. Insurance is also accepted.
AdditionalCosts	number		>= 0	Yes	Other customs costs; send 0 when none apply.
Importer	object			Yes	Importer details.
Importer.CompanyName	text			Yes	Importer company or recipient name.
Importer.Contact Name	text			No	Importer contact name.
Importer.Address 1	text			Yes	Primary importer address line.
Importer.Address 2	text			No	Additional importer address line.
Importer.City	text			Yes	Importer city.
Importer.State	text			No	Importer state or region.
Importer.Postcode	text			Yes	Importer postal code.
Importer.Country Code	text			Yes	Importer country code.
Importer.Phone	text			No	Importer phone number.

Importer.Email	text			No	Importer email address.
Details	array		non-empty	Yes	Declared goods.
Details[].Description	text			Yes	Product description.
Details[].UnitCode	text			No	Product unit code.
Details[].HsCode	text			Yes	Commodity/HS code.
Details[].UnitValue	number		>= 0	Yes	Value per unit.
Details[].Quantity	number		> 0	Yes	Number of units.
Details[].CountryOfManufacture	text			Yes	Country of manufacture.
Details[].NetUnitWeight	number		>= 0	Yes	Net weight per unit.
Details[].GrossUnitWeight	number		>= 0	Yes	Gross weight per unit.

5.7. The format of address for deliveries to door (to client's address)

JSON data-key: 'shipping'

Note: A shipping object is required by Courier v2 validation for every order. The selected courier determines field-level requirements.

Variable	Type	Max_length	Value	Required	Explanation
ship_name*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Name of the person to whom the order will be shipped to
ship_surname*	text	Venipak – 30 DPD – 40 In_Post - 30		Courier-specific*	Surname of the person to whom the order will be shipped to

postcode	number	Venipak – 5 DPD – 5 In_Post - 5		Courier-specific*	The number of post code digits depends on consignee country: EE – 5 digits LT – 5 digits LV – 4 digits PL – 5 digits RO – 6 digits
street**	text	Venipak – 50 DPD – 40 In_Post - 50		Courier-specific*	Street
building**	text	Venipak – 30 DPD – 30 In_Post - 30		Courier-specific*	House number
flat**	text	Venipak – 50 DPD – 50 In_Post - 50		Courier-specific*	Apartment (flat) number
city	text	Venipak – 40 DPD – 40 In_Post - 40		Courier-specific*	City
phone	number	Venipak – 20 DPD – 20 In_Post - 9		Courier-specific*	Phone of the person to whom the order will be shipped to. Prefix is mandatory (LT: +370, LV: +371, EE: +372...)

* Shipping-name field requirements and length limits are courier-specific. Send both values when available.

** Shipping-address field requirements and combined length limits are courier-specific. Send street, building, flat, city, postcode and phone as applicable to the selected courier.

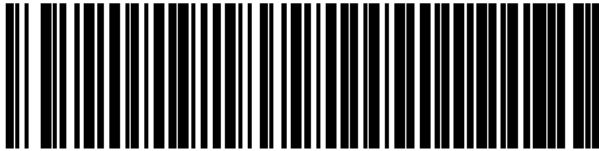
Note: if identical, ship_name, ship_surname and phone can be duplicated from pick up person data.

5.8. Generating label

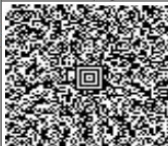
Labels are returned Base64-encoded. The process response contains a courier label in file (a single value or an array, depending on courier) and EshopWedrop tracking/label data in tracking_numbers. Print/apply the labels required by the selected courier.

Visual example of courier labels is visible below:

5.10.1 Venipak label:

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 label:

test test street 1-1 65548 city LT LIETUVA		DPD Lietuva Ulpianko g. 161 LT-02101 Vilnius PVA: +370649077 Faksas: +3706410614 PVA: LT11032917 089 Ulpianko g. 160 02121 Vilnius Lietuva Tel.: +3709999999
Phone 691234567 Info P000013231;comment MPSID 0580902319598320211028	Pakuotų ki... 1/1 Svoris 1.000	
Parcel ref. 11132-P000013231 0 Shipment ref. 11132-P000013231		
0580 9023 1959 83 X D-B2C Siuntos nr.: Pasiuaga LT-DPD-0592 327-LT-65548 944 2021-10-27 15:47:55 21090605 WebManifest v1.0.0  0065 5480 5809 0231 9598 3327 4400		

5.10.3. InPost label:

Rejon kurierski	Mikrorejon	Sortownia	Stacja doręczająca
D20		LO20	WA2
 520000016905800014698202			
Numer paczki: 520000016905800014698202		1/1	
Odbiorca:	test test street 1-1 05-800 Pruszków 691234567 test test	Typ paczki: Paczka Wysokość: 28.00 cm Długość: 31.00 cm Szerokość: 18.00 cm Waga rzeczywista: 10.00 kg Waga gabarytowa: 2.60 kg	
Nadawca:	EshopWedrop Poland Ul. 3 Maja Budynek A3 Pruszków 05-800 Dmytro Borishchuk 736625289	WERYFIKACJA WAGI ☐ UBEZPIECZENIE: 0.00 POBRANIE: 0,00 PLN Data wydruku: 2021-10-27 15:18 NADANIE W PACZKOMACIE ☐ Standard MPK:	
Płatnik:	ESHOPWEDROP POLAND		
Uwagi:	Order ID:11141		
Administratorem danych osobowych jest InPost Sp. z o.o. z siedzibą ul. Wileńska 28, 35-552 Kraków. Więcej informacji na inpost.pl			

5.9. Example of the POST Request:

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. Example of the Response:

```

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

```

5.11. Error notification:

```

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

```

Note: In case error is happening on customs declaration, then the entire order (all parcels) will be canceled.

6. Cancel an order

This method allows to cancel the order registered in the request.

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

Method: **GET**

Data: **order_id** – unique internal number provided by client during order registration through API

Success result example:

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

Unsuccessful result example:

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

7. Validate order for shipment

This method will validate the order data of the request and return success if the order passes all the restrictions.

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

Method: **POST**

Data: Validation method uses the same request data as the order registration method (section 5).

Success result example:

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

Unsuccessful result example:

```
{
```

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

8. Get Destinations list

This method will return all the available destinations for the requested API user.

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

Method: **GET**

Data: **origins_id** – origin ID must be provided to get available list of destinations.

Success result example:

```
{
  "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"}
  ]
}
```

Unsuccessful result example:

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

9. Get Origins list

This method will return all the available origin countries for the requested API user.

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

Method: **GET**

Data: -

Success result example:

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

Unsuccessful result example:

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

10. Get list of available delivery methods

This method will return all the available last mile couriers and their delivery methods for the requested destination.

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

Method: **GET**

Data: **destinations_id** – destination for which the delivery methods should be returned

Success result example:

```
{
  "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"}
  }
}
```

```
}}
```

Unsuccessful result example:

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

11. Get order status

This method returns the status-history events recorded for the requested parcel.

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

Method: **GET**

Data: parcel_id – ID of the parcel from the client's system (section 5.8).

Success result example:

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

Unsuccessful result example:

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

Historical status labels are returned dynamically. The list below is retained as a legacy reference and may not contain every event.

Status	Explanation
Ready for dispatch	<i>Order was registered by client via API</i>
Incoming to origin warehouse	<i>Order is in transit to origin warehouse (OW)</i>
In origin warehouse	<i>Order has arrived to origin warehouse (OW)</i>
Prepared for loading	<i>Order prepared for loading to the truck in OW</i>
In transit	<i>Order left the origin warehouse</i>
Arrived to destination	<i>Order was scanned in the Destination warehouse (DW)</i>
Sent through courier	<i>Order pick-ed up from DW by last mile courier</i>
Waiting for pick-up	<i>Order is waiting to be collected by customer from the warehouse</i>
Completed	<i>Order delivered to the customer</i>
Lost	<i>Order was lost</i>
Cancelled	<i>Order was cancelled</i>

12. Generate shipment manifest

This method will generate a manifest of all orders registered via API based on the date intervals provided in the request.

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

Method: **POST**

Data:

destinations_id - destination country ID of the manifest;

origins_id – origin country ID of the manifest;

date_from(date) – orders registered from this date will be included in the manifest;

date_to(date) – orders registered to this date will be included in the manifest;

sender_address(varchar(255)) – sender's address from where the orders will be picked-up.

Request example:

```
{
  "destinations_id": [1],
```

```
"origins_id": 2,  
"date_from": "2020-09-22",  
"date_to": "2020-09-24",  
"sender_address": "test address"  
}
```

If manifest is generated for parcels of several destinations, please list all destination IDs:

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

Success result example:

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

Unsuccessful result example:

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

Example of the PDF manifest:



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		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				Volume
					Length	Width	Height	Actual weight	
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. Download manifest

This method will return the base64 encoded file of previously generated manifest.

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

Method: **GET**

Data: barcode – manifest barcode returned by the generation response.

Success result example:

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

Unsuccessful result example:

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

14. Order tracking

Track & Trace links to clients are listed below. These links can be provided to clients buyers, so users can go and check the status of their shipment themselves.

Lithuania:

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

Latvia:

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

Estonia:

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

Poland:

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

Note 1: If client wants to provide tracking link to their buyers, it is possible to merge tracking link with tracking code, so client does not have to copy tracking code manually. Example of merged link:

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

Explanation:

<https://eshopwedrop.lt/delivery-options-lt> - link to a local website

?parcel=1029032 - tracking code of parcel

#block-track - positioning for automated scroll down

15. Get label

This method will return the base64 encoded file of previously generated label.

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

Method: **GET**

Data: tracking_number - order or parcel tracking number; file_type - pdf or zpl.

Success result example:

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

Unsuccessful result example:

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

16. Get additional services

This method returns the additional-service configuration for the selected destination and delivery method.

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

Method: GET

Data: destinations_id - destination ID returned by Get destinations list; pickup_modules_id - delivery-method ID returned by Get list of available delivery methods.

Success result example:

```
{
```

```

"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"
}
}

```

Unsuccessful result example:

```

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

```

17. Get click & collect points

This method returns available click-and-collect pickup points for the selected destination and delivery method.

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

Method: GET

Data: destinations_id – destination ID returned by Get destinations list; pickup_modules_id – delivery-method ID returned by Get list of available delivery methods. Pickup-point fields are courier-specific.

Success result example:

```

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

```

Unsuccessful result example:

```

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

```

18. Close transport batch

This method closes a transport batch, creates the transport manifest and returns the generated documents.

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

Method: POST

Data:

Variable	Type	Max_length	Value	Required	Explanation
batch_reference	text	191	BATCH-20260715-001	Yes	Unique reference for this transport batch. Repeating the exact same request is idempotent; the same reference with different data returns an error.
cmr_truck_plate_number	text	64	B123ABC	Yes	Truck registration/CMR plate number.
origins_id	number		14	Yes	Origin ID returned by Get origins list. Every order in order_ids must belong to this origin.
sender_address	text		Warehouse street 1, Vilnius	Yes	Sender address printed on the generated transport documents.
closed_at	datetime		2026-07-15 14:00:00	No	Batch closing time in the origin time zone, format YYYY-MM-DD HH:mm:ss. If omitted, the current server time is used.
order_ids	array of text / number		["01111111", "011111112"]	Yes	External order IDs. Orders must belong to the authenticated customer, be ready for dispatch, be unmanifested, and not already belong to a batch.

Example request data:

```
{
  "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": ["01111111", "011111112"]
}
```

Success result example:

```
{
```

```

"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}"
  }
}
}

```

Unsuccessful result example:

```

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

```

19. Get transport batch

This method returns batch metadata and generated-document metadata. It does not include document content.

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

Method: GET

Data: batch_reference – transport-batch reference supplied when closing the batch.

Success result example:

```

{
  "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"
    }
  }
}

```

Unsuccessful result example:

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

20. Download transport batch document

This method returns a generated transport-batch document as base64 content.

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

Method: GET

Data: batch_reference – transport-batch reference; document_type – commercial_invoice, packing_list, import_manifest, export_manifest, transport_manifest, or one of these with a country suffix (for example, commercial_invoice_GB).

Success result example:

```
{
  "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}"
  }
}
```

Unsuccessful result example:

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

21. Additional information

8.4 If C.O.D. (Cash on Delivery) will be used, all amounts must be converted to a currency of client's country. These amounts will be collected from clients by couriers.

8.5 If any error occurs, the response will return status FALSE and an error message explanation.

8.6 Our web service is built around REST base architecture, meaning HTTP Protocol is used to exchange data between applications and systems. All the data provided in the exchange should be formatted to JSON.

8.7 Character-encoding: UTF-8

8.8 For optional data, you can either send it as value 0, meaning FALSE, or remove the key completely from the request.

8.9 IT contacts for integration purposes and integration support:

Email: Support@eshopwedrop.com 	
---	--