



---

# ***HP Shipping Service API***

***Technical documentation***

---

## Table of contents

1. Change log .....	4
2. Getting started.....	5
3. Authorization – DX_Security_WebAPI .....	6
Response parameters.....	6
4. API URL-s.....	7
5. Shipment methods .....	8
5.1. CreateShipmentOrders .....	8
Definition .....	8
Request parameters .....	8
Request example .....	11
Response example.....	13
Response parameters.....	14
5.2. CancelShipmentOrders .....	14
Definition .....	14
Request parameters .....	14
Request example .....	15
Response example.....	15
Response parameters.....	15
5.3. GetShipmentStatus.....	16
Definition .....	16
Request parameters .....	16
Request example .....	16
Response example.....	16
Response parameters.....	17
5.4. GetShippingLabels.....	18
Definition .....	18
Request parameters .....	18
Request example .....	18
Response example.....	19
Response parameters.....	19
6. Delivery point methods .....	20
6.1. GetParcelDeliveryPoint .....	20

Definition .....	20
Request parameters .....	20
Request example .....	20
Response example .....	21
Appendix A – SCANS - STATUS AND REASONS FOR PARCEL COLLECTION NOT DONE AND NON-DELIVERY .....	23
<b>STATUS</b> .....	23
REASONS OF PARCEL COLLECTION IS NOT DONE (NPR) .....	23
REASONS OF NON-DELIVERY (IND) .....	23
Appendix B – List of comments on scan NPA .....	25
Appendix C – Examples of calling methods C# i JS and PHP-cURL .....	28

## 1. Change log

Change No.	Since	Description
<b>1</b>	28.11.2022.	Initial version

## 2. Getting started

This DXWebAPI service enables contract users of Hrvatska pošta d.d. to:

- create shipment orders
- cancel shipment orders
- get shipment status
- get shipping labels
- get parcel delivery point

Before using this DXWebAPI service you must be contract user of Hrvatska pošta d.d..  
If you don't have the required login credentials please contact Hrvatska pošta d.d..

DXWebAPI service communication is designed for using via **HTTPS**.

DXWebAPI service supports approach: **REST** (format **JSON**).

Used HTTP response codes: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Status/>



#### 4. API URL-s

OKRUŽENJE	URL
<b>TEST - DX_Security_WebAPI</b>	<a href="https://dxwebapit.posta.hr:9000/api/authentication/client_auth">https://dxwebapit.posta.hr:9000/api/authentication/client_auth</a>
<b>PRODUCTION - DX_Security_WebAPI</b>	<a href="https://dxwebapi.posta.hr:9000/api/authentication/client_auth">https://dxwebapi.posta.hr:9000/api/authentication/client_auth</a>
<b>TEST – DXWebAPI</b>	<a href="https://dxwebapit.posta.hr:9020/api/{format}/{methodName}">https://dxwebapit.posta.hr:9020/api/{format}/{methodName}</a>
<b>PRODUCTION – DXWebAPI</b>	<a href="https://dxwebapi.posta.hr:9020/api/{format}/{methodName}">https://dxwebapi.posta.hr:9020/api/{format}/{methodName}</a>

## 5. Shipment methods

### 5.1. CreateShipmentOrders

#### Definition

This method is used for creating shipment orders.

Method	URL example - TEST
POST	https://dxwebapit.posta.hr:9020/api/shipment/create_shipment_orders

#### Request parameters

#### Class ShipmentOrdersData

Parameter	Type	Description	REQUIRED
<b>parcels</b>	List< <a href="#">ShipmentOrderData</a> >	A list of shipment orders to create	YES
<b>return_address_label</b>	Bool	Flag for shipping labels in a response for created shipment orders in PDF format <u>Possible values:</u> <b>false</b> – NO <b>true</b> – YES	

#### Class ShipmentOrderData

Parameter	Type	Description	REQUIRED
<b>client_reference_number</b>	String	Unique client ID of shipment order Max. length: 100	YES
<b>service</b>	Integer	Type of service <u>Possible values:</u> <b>24</b> – Pallet shipment <b>26</b> – Package 24 D+1 <b>29</b> – Package 24 D+2 <b>32</b> – Package 24 D+3 <b>38</b> – Package 24 D+4	YES
<b>payed_by</b>	Integer	Shipment payer <u>Possible values:</u> <b>1</b> – Sender <b>2</b> – Recipient If NULL Sender is default.	
<b>delivery_type</b>	Integer	Shipment delivery type <u>Possible values:</u> <b>1</b> – Address <b>2</b> – Post office <b>3</b> – Parcel locker	YES
<b>value</b>	Decimal	Insured shipment value Number of decimal places: 2	



<b>payment_value</b>	Decimal	Cash-on-delivery amount Number of decimal places: 2	YES if additional service 9 is chosen (Cash-on-delivery shipment)
<b>pickup_type</b>	Integer	Shipment pickup type <u>Possible values:</u> <b>1</b> – Address <b>2</b> – Post office <b>3</b> – Parcel locker	
<b>parcel_size</b>	String	Parcel size <u>Possible values:</u> <b>X</b> – Parcel dimension XS <b>S</b> – Parcel dimension S <b>M</b> – Parcel dimension M <b>L</b> – Parcel dimension L Parcel dimensions: <a href="https://www.posta.hr/dimenzije-pretinaca/9279">https://www.posta.hr/dimenzije-pretinaca/9279</a>	YES if delivery_type 3 is chosen (Shipment is delivered to a Parcel locker)
<b>reference_field_B</b>	String	Order reference field B NOT UNIQUE. Max. length: 100	
<b>reference_field_C</b>	String	Order reference field C NOT UNIQUE. Max. length: 100	
<b>reference_field_D</b>	String	Order reference field D NOT UNIQUE. Max. length: 100	
<b>sender</b>	<a href="#">SenderData</a>	Sender's data	YES
<b>recipient</b>	<a href="#">RecipientData</a>	Recipient's data	YES
<b>additional_services</b>	List< <a href="#">AdditionalServiceData</a> >	Additional services	
<b>packages</b>	List< <a href="#">PackageData</a> >	Package list	YES

## SenderData

Parameter	Type	Description	REQUIRED
<b>sender_name</b>	String	Sender's name (person or organization) Max. length: 50	YES
<b>sender_phone</b>	String	Sender's phone number Max. length: 25	YES
<b>sender_email</b>	String	Sender's email address Max. length: 25	
<b>sender_street</b>	String	Sender's name of the street Max. length: 50	YES
<b>sender_hnum</b>	String	Sender's number of the house Max. length: 10	YES
<b>sender_hnum_suffix</b>	String	Sender's suffix number of the house Max. length: 10	YES
<b>sender_zip</b>	String	Sender's area zip code Max. length: 5	YES
<b>sender_city</b>	String	Sender's town or village name Max. length: 25	YES
<b>sender_pickup_center</b>	String	Sender's post office code or parcel locker for pickup Max. length: 5	YES if pickup_type 2 or 3 is chosen

## RecipientData

Parameter	Type	Description	REQUIRED
<b>recipient_name</b>	String	Recipient's name (person or organization) Max. length: 50	YES
<b>recipient_phone</b>	String	Recipient's phone number Max. length: 25	YES
<b>recipient_email</b>	String	Recipient's email address Max. length: 25	
<b>recipient_street</b>	String	Recipient's name of the street Max. length: 50	YES
<b>recipient_hnum</b>	String	Recipient's number of the house Max. length: 10	YES
<b>recipient_hnum_suffix</b>	String	Recipient's suffix number of the house Max. length: 10	YES
<b>recipient_zip</b>	String	Recipient's area zip code Max. length: 5	YES
<b>recipient_city</b>	String	Recipient's name of the town or village Max. length: 25	YES
<b>recipient_delivery_center</b>	String	Recipient's code of post office or parcel locker for delivery Max. length: 5	YES if delivery_type 2 or 3 is chosen

## Class AdditionalServiceData

Parameter	Type	Description	REQUIRED
<b>additional_service_id</b>	Integer	ID of additional service <u>Possible values:</u> <b>1</b> – Personal delivery <b>3</b> – Delivery on Saturdays <b>4</b> – With the advice of receipt (AR) <b>9</b> – Cash-on-delivery shipment <b>30</b> – Send a notification to the recipient <b>32</b> – Send an e-mail to the recipient <b>47</b> – Consolidated shipment	

## Class PackageData

Parameter	Type	Description	REQUIRED
<b>barcode</b>	String	Barcode given by Hrvatska pošta d.d. (before creating shipment order) Max. length: 20	YES if barcode_type 0 is chosen
<b>barcode_type</b>	Integer	Barcode type <u>Possible values:</u> <b>0</b> – Barcode given by Hrvatska pošta d.d. (before creating shipment order) <b>1</b> – Barcode given from response after creating shipment order	YES
<b>barcode_client</b>	String	Client's own barcode Max. length: 500	YES if barcode_type 1 is chosen

<b>weight</b>	Decimal	Package weight in kilograms Number of decimal places: 3	YES
---------------	---------	--	-----

### Request example

```
{
  "parcels": [
    {
      "client_reference_number": "test",
      "service": "26",
      "payed_by": 1,
      "delivery_type": 1,
      "value": null,
      "payment_value": 20,
      "pickup_type": 1,
      "parcel_size": "X",
      "reference_field_B": "string",
      "reference_field_C": "string",
      "reference_field_D": "string",
      "sender": {
        "sender_name": "string",
        "sender_phone": "string",
        "sender_email": "string",
        "sender_street": "string",
        "sender_hnum": "string",
        "sender_hnum_suffix": "string",
        "sender_zip": "string",
        "sender_city": "string",
        "sender_pick_up_center": "string"
      },
      "recipient": {
        "recipient_name": "string",
        "recipient_phone": "string",
        "recipient_email": "string",
        "recipient_street": "string",
        "recipient_hnum": "string",
        "recipient_hnum_suffix": "",
        "recipient_zip": "string",
        "recipient_city": "string",
        "recipient_delivery_center": "string"
      },
      "additional_services": [
        {
          "additional_service_id": 9
        }
      ],
      "packages": [
        {
          "barcode": "",
          "barcode_type": 1,
          "barcode_client": "string",
          "weight": 3
        },
        {
          "barcode": "",

```

```

        "barcode_type": 1,
        "barcode_client": "string",
        "weight": 3
    }
]
},
{
    "client_reference_number": "test1",
    "service": "26",
    "payed_by": 1,
    "delivery_type": 1,
    "value": null,
    "payment_value": 0,
    "pickup_type": 1,
    "parcel_size": "S",
    "reference_field_B": "string",
    "reference_field_C": "string",
    "reference_field_D": "string",
    "sender": {
        "sender_name": "string",
        "sender_phone": "string",
        "sender_email": "string",
        "sender_street": "string",
        "sender_hnum": "string",
        "sender_hnum_suffix": "string",
        "sender_zip": "string",
        "sender_city": "string",
        "sender_pick_up_center": "string"
    },
    "recipient": {
        "recipient_name": "string",
        "recipient_phone": "string",
        "recipient_email": "string",
        "recipient_street": "string",
        "recipient_hnum": "string",
        "recipient_hnum_suffix": "string",
        "recipient_zip": "string",
        "recipient_city": "string",
        "recipient_delivery_center": "string"
    },
    "additional_services": [
        {
            "additional_service_id": 30
        }
    ],
    "packages": [
        {
            "barcode": "",
            "barcode_type": 1,
            "barcode_client": "string",
            "weight": 3
        },
        {

```

```

        "barcode": "",
        "barcode_type": 1,
        "barcode_client": "string",
        "weight": 3
    }
]
},
"return_address_label": true
}

```

## Response example

```

{
  "ShipmentOrdersList": [
    {
      "ClientReferenceNumber": "test",
      "ResponseStatus": 0,
      "Packages": [
        {
          "barcode": "LE900002111HR",
          "barcode_type": 1,
          "barcode_client": "test12",
          "weight": 3.0
        },
        {
          "barcode": "LE900002125HR",
          "barcode_type": 1,
          "barcode_client": "test22",
          "weight": 3.0
        }
      ],
      "ErrorCode": null,
      "ErrorMessage": null,
      "MessageID": ""
    },
    {
      "ClientReferenceNumber": "test1",
      "ResponseStatus": 0,
      "Packages": [
        {
          "barcode": "LE900002139HR",
          "barcode_type": 1,
          "barcode_client": "test12",
          "weight": 3.0
        },
        {
          "barcode": "LE900002142HR",
          "barcode_type": 1,
          "barcode_client": "test22",
          "weight": 3.0
        }
      ]
    }
  ]
}

```

```

    ],
    "ErrorCode": null,
    "ErrorMessage": null,
    "MessageID": ""
  }
],
"ShipmentsLabel": "base64string"
}

```

## Response parameters

### Class CreateShipmentOrdersMainResponse

Parameter	Type	Description
<b>ShipmentOrdersList</b>	List<CreateShipmentOrdersResponse>	List of created or rejected shipment orders depending of potential errors
<b>ShipmentsLabel</b>	String	Shipping labels for created shipment orders in PDF format

### Class CreateShipmentOrdersResponse

Parameter	Type	Description
<b>ClientReferenceNumber</b>	String	Unique client ID of shipment order
<b>ResponseStatus</b>	EnumResponseStatus	<b>0</b> – OK <b>1</b> – ERROR
<b>Packages</b>	List<PackageData>	List of created packages
<b>ErrorCode</b>	Integer	Error code
<b>ErrorMessage</b>	String	Error message
<b>MessageID</b>	String	Error ID

## 5.2. CancelShipmentOrders

### Definition

This method is used to cancel shipping orders.

Method	URL example - TEST
POST	<a href="https://dxwebapit.posta.hr:9020/api/shipment/cancel_shipment_orders">https://dxwebapit.posta.hr:9020/api/shipment/cancel_shipment_orders</a>

## Request parameters

### Class CancelOrdersData

Parameter	Type	Description	REQUIRED
<b>parcels</b>	List< <a href="#">CancelOrderData</a> >	List of shipment orders to cancel	YES

## Class CancelOrderData

Parameter	Type	Description	REQUIRED
client_reference_number	String	Unique client ID of shipment order Max. length: 100	YES

### Request example

```
{
  "parcels": [
    {
      "client_reference_number": "test"
    },
    {
      "client_reference_number": "test1"
    }
  ]
}
```

### Response example

```
[
  {
    "ClientReferenceNumber": "test",
    "ResponseStatus": 0,
    "ErrorCode": null,
    "ErrorMessage": null
  },
  {
    "ClientReferenceNumber": "test1",
    "ResponseStatus": 0,
    "ErrorCode": null,
    "ErrorMessage": null
  }
]
```

### Response parameters

#### Class CancelShipmentOrdersResponse

Parameter	Type	Description
ClientReferenceNumber	String	Unique client ID of shipment order
ResponseStatus	EnumResponseStatus	<b>0</b> – OK <b>1</b> – ERROR
ErrorCode	Integer	Error code
ErrorMessage	String	Error message

### 5.3. GetShipmentStatus

#### Definition

This method is used to retrieve shipping orders status.

Method	URL example - TEST
GET	https://dxwebapit.posta.hr:9020/api/shipment/get_shipment_status

#### Request parameters

##### Class BarcodesData

Parameter	Type	Description	REQUIRED
<b>barcodes</b>	List< <a href="#">BarcodeData</a> >	List of barcodes to retrieve status	YES

##### Class BarcodeData

Parameter	Type	Description	REQUIRED
<b>barcode</b>	String	Package barcode to retrieve status Max. length: 20	YES

#### Request example

```
{  "barcodes": [
    {
      "barcode": "LE347087966HR"
    },
    {
      "barcode": "LE347087952HR"
    }
  ]
}
```

#### Response example

```
[
  {
    "Barcode": "LE347087966HR",
    "PackageScansList": [
      {
        "Scan": "NOV",
        "ScanDescription": "Definirani podaci o pošiljci",
        "ScanTime": "2022-11-16T11:15:33.257",
        "Center": "ROOT",
        "Comment": null,
        "CommentNPA": "5000002"
      }
    ],
    "ResponseStatus": 0,
    "ErrorCode": null,
  }
]
```



```

    "ErrorMessage": null
  },
  {
    "Barcode": "LE347087952HR",
    "PackageScansList": [
      {
        "Scan": "NOV",
        "ScanDescription": "Definirani podaci o pošiljci",
        "ScanTime": "2022-11-16T11:30:22.763",
        "Center": "ROOT",
        "Comment": null,
        "CommentNPA": "5000002"
      }
    ],
    "ResponseStatus": 0,
    "ErrorCode": null,
    "ErrorMessage": null
  }
]

```

## Response parameters

### Class GetShipmentStatusResponse

Parameter	Type	Description
<b>Barcode</b>	String	Package barcode
<b>PackageScansList</b>	List< <a href="#">PackageScanData</a> >	List of package scans
<b>ResponseStatus</b>	EnumResponseStatus	<b>0</b> – OK <b>1</b> – ERROR
<b>ErrorCode</b>	Integer	Error code
<b>ErrorMessage</b>	String	Error message

### Class PackageScanData

Parameter	Type	Description
<b>Scan</b>	String	Scan code <b>List of scans:</b> <a href="#">Appendix A</a>
<b>ScanDescription</b>	String	Scan description
<b>ScanTime</b>	DateTime	Scan date and time
<b>Center</b>	String	Center in which the scan was made
<b>Comment</b>	String	Comment on scan
<b>CommentNPA</b>	String	Comment on scan NPA <b>List of comments:</b> <a href="#">Appendix B</a>

## 5.4. GetShippingLabels

### Definition

This method is used for retrieval of package address labels.

It is possible to retrieve the address label by parcel barcodes **and/or** by client order number.

Method	URL example - TEST
GET	https://dxwebapit.posta.hr:9020/api/shipment/get_shipping_labels

### Request parameters

#### Class LabelsData

Parameter	Type	Description	REQUIRED
<b>client_reference_number</b>	String	Unique client ID of shipment order Max. length: 100	YES if barcodes is NULL or EMPTY
<b>barcodes</b>	List< <a href="#">BarcodeData</a> >	List of barcodes to retrieve labels	YES if CRN is NULL or EMPTY
<b>A4</b>	Bool	Flag for A4 paper size (4 labels on 1 page) <u>Possible values:</u> <b>false</b> – NO <b>true</b> – YES If NULL 0 is default.	

#### Class BarcodeData

Parameter	Type	Description	REQUIRED
<b>barcode</b>	String	Package barcode Max. length: 20	

### Request example

```
{
  "client_reference_number": "test",
  "barcodes": [
    {
      "barcode": "LE900001297HR"
    },
    {
      "barcode": "LE900001310HR"
    }
  ],
  "A4": false
}
```

## Response example

```
{
  "PackageLabel": "base64string",
  "ResponseStatus": 0,
  "ErrorCode": null,
  "ErrorMessage": null
}
```

## Response parameters

### Class GetShippingLabelsResponse

Parameter	Type	Description
<b>PackageLabel</b>	String	Package labels of created shipment orders in „base64“ string format
<b>ResponseStatus</b>	EnumResponseStatus	<b>0</b> – OK <b>1</b> – ERROR
<b>ErrorCode</b>	Integer	Error code
<b>ErrorMessage</b>	String	Error message

## 6. Delivery point methods

### 6.1. GetParcelDeliveryPoint

#### Definition

This method is used to retrieve accurate and available information about working hours, addresses and other information of Croatian Post offices and parcel lockers.

Method	URL example - TEST
GET	https://dxwebapit.posta.hr:9020/api/delivery_point/get_parcel_delivery_point

#### Request parameters

##### Class ParcelDeliveryPointData

Parameter	Type	Description	REQUIRED
<b>facilityType</b>	String	Parameter that filters data on post offices and parcel lockers <u>Possible values:</u> <b>ALL</b> – Retrieves post offices and parcel lockers <b>PU</b> – Retrieves only post lockers <b>PAK</b> – Retrieves only parcel lockers	YES
<b>nextWeek</b>	Integer	A parameter that determines whether the working hours for post offices will be retrieved for the current week or for the current and the following week <u>Possible values:</u> <b>0</b> – Retrieves post office working hours for the current week <b>1</b> – Retrieves post office working hours for the current and the following week	YES
<b>searchText</b>	String	A parameter that filters the retrieved data by name, location, postal code or parcel locker code and street Empty string does not filter results	YES

#### Request example

1. Example of a service call that retrieves all post offices and parcel lockers with business hours for the current week, without filtering the results

```
{  
  "facilityType": "ALL",  
  "nextWeek": 0,  
  "searchText": ""  
}
```

## 2. Example of a service call that retrieves only parcel lockers

```
{
  "facilityType": "PAK",
  "nextWeek": 0,
  "searchText": ""
}
```

## 3. Example of a service call that retrieves only post offices with business hours for the current and following week

```
{
  "facilityType": "PU",
  "nextWeek": 1,
  "searchText": ""
}
```

## 4. Example of a service call that retrieves data by filtering

```
{
  "facilityType": "ALL",
  "nextWeek": 0,
  "searchText": "62102"
}
{
  "facilityType": "PU",
  "nextWeek": 0,
  "searchText": "Zagreb"
}
```

## Response example

```
"postaInfoList":
[
  {
    "type": "string",
    "name": "string",
    "zip": "string",
    "city": "string",
    "address": "string",
    "tel": "string",
    "fax": "string",
    "geoLatitude": 0,
    "getLongitude": 0,
    "workingHours":
    [
      {
        "weekDay": 0,
        "date": "string",
        "openingTime":
        {
          "hour": 0,
          "minute": 0,
          "second": 0
        },
      },
    ],
  },
]
```

```

        "closingTime":
        {
            "hour": 0,
            "minute": 0,
            "second": 0
        },
        "closed": "string"
    }
]
},
"paketomatInfoList":
[
    {
        "type": "string",
        "name": "string",
        "zip": "string",
        "city": "string",
        "address": "string",
        "geoLat": 0,
        "getLng": 0,
        "code": "string"
    }
]
}

```

The service returns two lists: one for post offices "postalInfoList" and one for parcel lockers "paketomatInfoList".

#### PostalInfoList

Within the list for post offices, the service returns information about working hours "workingHours" structured in such a way that working hours are returned separately for each day of the week.

WorkingHours contains the following parameters within its structure:

Parameter	Type	Description
<b>weekDay</b>	Integer	Brojčana oznaka dana u tjednu 1 – ponedjeljak 2 – utorak . . 7 - nedjelja
<b>date</b>	String	Designates the date
<b>openingTime</b>	String	Designates the opening time of the postal office
<b>closingTime</b>	String	Designates the closing time of the postal office
<b>Closed</b>	String	Designates whether the postal office is open or closed 0 – PO closed 1 – PO open

## Appendix A – SCANS - STATUS AND REASONS FOR PARCEL COLLECTION NOT DONE AND NON-DELIVERY

### STATUS

- NOV – new parcel
- ZAP – debit courier for parcel collection
- NPR – parcel collection is not done
- PRI – parcel collection is done
- ZPR – receipt of the parcel into the system-scanned in central HUB
- OTP – parcel transit between post depos
- ZOT – receipt of the parcel from other post depo
- ZAD – debit courier for parcel delivery
- IOD – delivery note in post depo-discharge of
- IND – non-delivery
- ODG – delivery delay for warehousing
- POV – parcel return
- IPP – finally delivery to the sender
- PRE – routed - change destination
- UKO – parcel loading to the container
- IZM – change parcel data

### REASONS OF PARCEL COLLECTION IS NOT DONE (NPR)

- 1050002 Recipient absent
- 1050006 Recipient absent
- 1050003 Unknown recipient
- 1050005 Parcel is not ready
- 1050007 Undoable
- 1050001 No parcel for collection
- 1050008 Sender info about collection attempt

### REASONS OF NON-DELIVERY (IND)

- 3900002 undeliverable
- 7010002 died
- 7010009 lost before delivery
- 7010010 lost after delivery
- 3900023 incomplete address
- 3900022 delivery agreed
- 3900025 second attempt delivery
- 3900004 refused by recipient
- 7010003 returned
- 3900027 the courier did not reach the address
- 3900017 recipient informed
- 7010004 for payment/delivery in post office
- 7010018 new attempt of delivery on order
- 3900007 recipient present, without money for payment
- 3900001 recipient absent
- 7010001 resend to

- 7010017 delivery in other post office
- 3900028 The user does not respond - informed
- 3900029 Wrong contact phone number of the user - notified
- 3900018 Package contents do not match
- 3900019 Damaged shipment
- 3900021 Issue at sorting center



## Appendix B – List of comments on scan NPA

id	code	name	group_code	event_id	name2
1	1050001	Parcels missing	NULL	8	Nerealiziran prikup
2	1050002	recipient absent	NULL	8	Nerealiziran prikup
3	1050003	unknown recipient	NULL	8	Nerealiziran prikup
4	1050004	Non standard shipment	NULL	8	Nerealiziran prikup
5	1050005	parcel is not ready	NULL	8	Nerealiziran prikup
6	1050006	recipient absent	NULL	8	Nerealiziran prikup
7	1050007	Not doable	NULL	8	Nerealiziran prikup
8	2300001	Parcel Shop closed	NULL	47	Unsuccessful handover to parcelShop
9	2300002	Parcel Shop full	NULL	47	Unsuccessful handover to parcelShop
10	2300003	Unsuccessful handover to parcel Shop	NULL	47	Unsuccessful handover to parcelShop
11	2303001	Parcel Shop full	NULL	50	Neuspješna predaja pošiljke u paketomat
12	2303002	Parcel Shop out of order	NULL	50	Neuspješna predaja pošiljke u paketomat
13	2303003	Unsuccessful handover to parcel Shop	NULL	50	Neuspješna predaja pošiljke u paketomat
14	2306001	Post office closed	NULL	53	Neuspješna predaja pošiljke u poštanski ured
15	2306002	Post office full	NULL	53	Neuspješna predaja pošiljke u poštanski ured
16	2306003	Unsuccessful handover to Post office	NULL	53	Neuspješna predaja pošiljke u poštanski ured
17	3900001	recipient absent	1	106	Pošiljka neuručiva
18	3900002	undeliverable	2	106	Pošiljka neuručiva
20	3900004	refused by recipient	2	106	Pošiljka neuručiva
23	3900007	recipient present, without money for payment	2	106	Pošiljka neuručiva
33	3900017	recipient informed	4	106	Pošiljka neuručiva
37	3900021	delivery sort error	8	106	Pošiljka neuručiva
38	3900022	delivery agreed	9	106	Pošiljka neuručiva
39	3900023	incomplete address	10	106	Pošiljka neuručiva
41	3900025	second attempt delivery	1	106	Pošiljka neuručiva
43	3900027	the courier did not reach the address	14	106	Pošiljka neuručiva
44	5000001	Defined shipment data in the call center	NULL	129	Definirani podaci o pošiljci
45	5000002	Defined shipment data through the call center	NULL	129	Definirani podaci o pošiljci

46	5000003	Defined shipment data through the EPK	NULL	129	Definirani podaci o pošiljci
47	5000004	Defined shipment data through the DP	NULL	129	Definirani podaci o pošiljci
48	7010001	resend to	NULL	134	MIPS Nadoslano u
49	7010002	died	NULL	134	MIPS Nadoslano u
50	7010003	returned	NULL	134	MIPS Nadoslano u
51	7010004	for payment/delivery in post office	NULL	134	MIPS Nadoslano u
52	7010005	for the district	NULL	134	MIPS Nadoslano u
53	7010006	For the postman	NULL	134	MIPS Nadoslano u
54	7010007	More obligated	NULL	134	MIPS Nadoslano u
55	7010008	Less obligated	NULL	134	MIPS Nadoslano u
56	7010009	lost before delivery	NULL	134	MIPS Nadoslano u
57	7010010	lost after delivery	NULL	134	MIPS Nadoslano u
58	7010011	Postage incorrectly charged	NULL	134	MIPS Nadoslano u
59	7010012	Defective money order/parcel	NULL	134	MIPS Nadoslano u
60	7010013	For military post	NULL	134	MIPS Nadoslano u
61	7010014	Disposition	NULL	134	MIPS Nadoslano u
62	7010015	the money order/shipment did not arrive	NULL	134	MIPS Nadoslano u
63	7010016	For post office box	NULL	134	MIPS Nadoslano u
64	7010017	delivery in other post office	NULL	134	MIPS Nadoslano u
65	7010018	new attempt of delivery on order	NULL	134	MIPS Nadoslano u
66	7010019	Request denied	NULL	134	MIPS Nadoslano u
67	1001001	Barcode scanned	NULL	11	Prikup pošiljaka kod pošiljatelja
68	1001002	Barcode not scanned	NULL	11	Prikup pošiljaka kod pošiljatelja
69	3900028	the user does not respond - informed	1	106	Pošiljka neuručiva
70	3900029	wrong contact phone number of the user - notified	1	106	Pošiljka neuručiva
71	5002001	Change of invoice type R1 - DP	NULL	1310	Ažuriranje pošiljke
72	5000005	Defined shipment data through CE	NULL	129	Definirani podaci o pošiljci
73	5000006	Defined shipment data on parcel Shop	NULL	129	Definirani podaci o pošiljci
74	5006001	The sender will store the parcel in the compartment of the parcels Shop	NULL	1320	Najava pošiljke paketomatu
75	5006002	The postman will store the package in the compartment of the parcel machine	NULL	1320	Najava pošiljke paketomatu

76	1080001	Free return	NULL	1320	Najava pošiljke paketomatu
78	1050008	Notified	NULL	8	Nerealiziran prikup
79	2210001	Return parcel delivered to parcel Shop	NULL	212	Pošiljka predana u paketomat
80	3900018	Parcel content mismatch	NULL	106	Pošiljka neuručiva
81	3900019	Parcel damaged	NULL	106	Pošiljka neuručiva
97	8010001	Parcel return	NULL	70	Utovar dostavnog vozila
98	2303004	The dimensions of the parcel do not fit in the package machine	NULL	50	Neuspješna predaja pošiljke u paketomat

## Appendix C – Examples of calling methods C# i JS and PHP-cURL

### 1.1 ClientAuth - authorization

#### C#

```
var client = new RestClient("https://dxwebapit.posta.hr:9000/api/authentication/client_auth");
client.Timeout = -1;
var request = new RestRequest(Method.POST);
request.AddHeader("Content-Type", "application/json");
var body = @"{
" + "\n" +
@"    ""username"": ""test"",
" + "\n" +
@"    ""password"": ""test""
" + "\n" +
@"}";
request.AddParameter("application/json", body, ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);
```

#### JS

```
var settings = {
  "url": "https://dxwebapit.posta.hr:9000/api/authentication/client_auth",
  "method": "POST",
  "timeout": 0,
  "headers": {
    "Content-Type": "application/json"
  },
  "data": JSON.stringify({
    "username": "test",
    "password": "test"
  }),
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

#### PHP – cURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => 'https://dxwebapit.posta.hr:9000/api/authentication/client_auth',
  CURLOPT_RETURNTRANSFER => true,
```

```

CURLOPT_ENCODING => '',
CURLOPT_MAXREDIRS => 10,
CURLOPT_TIMEOUT => 0,
CURLOPT_FOLLOWLOCATION => true,
CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
CURLOPT_CUSTOMREQUEST => 'POST',
CURLOPT_POSTFIELDS => '{
    "username": "test",
    "password": "test"
}',
CURLOPT_HTTPHEADER => array(
    'Content-Type: application/json'
),
));

$response = curl_exec($curl);
curl_close($curl);
echo $response;

```

## 2.1 CreateShipmentOrders

### C#

```

var client = new RestClient("https://dxwebapit.posta.hr:9020/api/shipment/create_shipment_orders");
client.Timeout = -1;
var request = new RestRequest(Method.POST);
request.AddHeader("Authorization", "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREU iOiI2NzU5ODQlLCJuYmYiOiJlMjNjM3MTM1MzQsImV4cCI6MTY2OTcyNzkyNCwiaWF0IjoxNjY5NzEzNTM0fQ.bilmFmhsbCADq8-GK7Apzrf35eyi3AMv0jiMZLLEgCI");
request.AddHeader("Content-Type", "application/json");
var body = @"{
" + "\n" +
@"    ""parcels"": [
" + "\n" +
@"{
" + "\n" +
@"""client_reference_number"": ""test"",
" + "\n" +
@"""service"": ""26"",
" + "\n" +
@"""payed_by"": 1,
" + "\n" +
@"""delivery_type"": 1,
" + "\n" +
@"""value"": null,
" + "\n" +
@"""payment_value"": null,
" + "\n" +
@"""pickup_type"": 1,
" + "\n" +
@"""parcel_size"": ""L"",
" + "\n" +

```

```

@""reference_field_B"": "",
" + "\n" +
@""reference_field_C"": "",
" + "\n" +
@""reference_field_D"": "",
" + "\n" +
@"      ""sender"":
" + "\n" +
@"      {
" + "\n" +
@"          ""sender_name"": ""testNameSender"",
" + "\n" +
@"          ""sender_phone"": ""+38598111111"",
" + "\n" +
@"          ""sender_email"": ""test@test.com"",
" + "\n" +
@"          ""sender_street"": ""testStreetSender"",
" + "\n" +
@"          ""sender_hnum"": ""1"",
" + "\n" +
@"          ""sender_hnum_suffix"": ""testHnumSuffixSender"",
" + "\n" +
@"          ""sender_zip"": 10000,
" + "\n" +
@"          ""sender_city"": ""Zagreb"",
" + "\n" +
@"          ""sender_pickup_center"": null
" + "\n" +
@"      },
" + "\n" +
@"      ""recipient"":
" + "\n" +
@"      {
" + "\n" +
@"          ""recipient_name"": ""testNameRecipient"",
" + "\n" +
@"          ""recipient_phone"": ""+385998274492"",
" + "\n" +
@"          ""recipient_email"": ""test@test.com"",
" + "\n" +
@"          ""recipient_street"": ""testStreetRecipient"",
" + "\n" +
@"          ""recipient_hnum"": 2,
" + "\n" +
@"          ""recipient_hnum_suffix"": ""testHnumSuffixRecipient"",
" + "\n" +
@"          ""recipient_zip"": 10000,
" + "\n" +
@"          ""recipient_city"": ""Zagreb"",
" + "\n" +
@"          ""recipient_delivery_center"": null
" + "\n" +
@"      },

```

```

" + "\n" +
@"      ""additional_services"": [
" + "\n" +
@"      {
" + "\n" +
@"          ""additional_service_id"": 1
" + "\n" +
@"      }
" + "\n" +
@"  ],
" + "\n" +
@"      ""packages"": [
" + "\n" +
@"      {
" + "\n" +
@"          ""barcode"": "",
" + "\n" +
@"          ""barcode_type"": 1,
" + "\n" +
@"          ""barcode_client"": ""test1"",
" + "\n" +
@"          ""weight"": 3
" + "\n" +
@"      },
" + "\n" +
@"      {
" + "\n" +
@"          ""barcode"": "",
" + "\n" +
@"          ""barcode_type"": 1,
" + "\n" +
@"          ""barcode_client"": ""test2"",
" + "\n" +
@"          ""weight"": 3
" + "\n" +
@"      }
" + "\n" +
@"  ]
" + "\n" +
@"}
" + "\n" +
@"],
" + "\n" +
@"      ""return_address_label"": true
" + "\n" +
@"}
" + "\n" +
@"
" + "\n" +
@"";
request.AddParameter("application/json", body, ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);

```

## JS

```
var settings = {
  "url": "https://dxwebapit.posta.hr:9020/api/shipment/create_shipment_orders",
  "method": "POST",
  "timeout": 0,
  "headers": {
    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2ZnU5ODQiLCJmFmhsbCADq8-GK7Apzrf35eyi3AMv0jIMZLLEgCI",
    "Content-Type": "application/json"
  },
  "data": JSON.stringify({
    "parcels": [
      {
        "client_reference_number": "test",
        "service": "26",
        "payed_by": 1,
        "delivery_type": 1,
        "value": null,
        "payment_value": null,
        "pickup_type": 1,
        "parcel_size": "L",
        "reference_field_B": "",
        "reference_field_C": "",
        "reference_field_D": "",
        "sender": {
          "sender_name": "testNameSender",
          "sender_phone": "+38598111111",
          "sender_email": "test@test.com",
          "sender_street": "testStreetSender",
          "sender_hnum": "1",
          "sender_hnum_suffix": "testHnumSuffixSender",
          "sender_zip": 10000,
          "sender_city": "Zagreb",
          "sender_pickup_center": null
        },
        "recipient": {
          "recipient_name": "testNameRecipient",
          "recipient_phone": "+385998274492",
          "recipient_email": "test@test.com",
          "recipient_street": "testStreetRecipient",
          "recipient_hnum": 2,
          "recipient_hnum_suffix": "testHnumSuffixRecipient",
          "recipient_zip": 10000,
          "recipient_city": "Zagreb",
          "recipient_delivery_center": null
        },
        "additional_services": [
          {
            "additional_service_id": 1
          }
        ]
      }
    ]
  })
}
```



```

        "packages": [
            {
                "barcode": "",
                "barcode_type": 1,
                "barcode_client": "test1",
                "weight": 3
            },
            {
                "barcode": "",
                "barcode_type": 1,
                "barcode_client": "test2",
                "weight": 3
            }
        ]
    },
    "return_address_label": true
}),
};

$.ajax(settings).done(function (response) {
    console.log(response);
});

```

## PHP – cURL

```

<?php

$curl = curl_init();

curl_setopt_array($curl, array(
    CURLOPT_URL => 'https://dxwebapit.posta.hr:9020/api/shipment/create_shipment_orders',
    CURLOPT_RETURNTRANSFER => true,
    CURLOPT_ENCODING => '',
    CURLOPT_MAXREDIRS => 10,
    CURLOPT_TIMEOUT => 0,
    CURLOPT_FOLLOWLOCATION => true,
    CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
    CURLOPT_CUSTOMREQUEST => 'POST',
    CURLOPT_POSTFIELDS => '{
        "parcels": [
            {
                "client_reference_number": "test",
                "service": "26",
                "payed_by": 1,
                "delivery_type": 1,
                "value": null,
                "payment_value": null,
                "pickup_type": 1,
                "parcel_size": "L",
                "reference_field_B": "",
                "reference_field_C": "",
                "reference_field_D": "",

```

```

"sender":
{
    "sender_name": "testNameSender",
    "sender_phone": "+38598111111",
    "sender_email": "test@test.com",
    "sender_street": "testStreetSender",
    "sender_hnum": "1",
    "sender_hnum_suffix": "testHnumSuffixSender",
    "sender_zip": 10000,
    "sender_city": "Zagreb",
    "sender_pickup_center": null
},
"recipient":
{
    "recipient_name": "testNameRecipient",
    "recipient_phone": "+385998274492",
    "recipient_email": "test@test.com",
    "recipient_street": "testStreetRecipient",
    "recipient_hnum": 2,
    "recipient_hnum_suffix": "testHnumSuffixRecipient",
    "recipient_zip": 10000,
    "recipient_city": "Zagreb",
    "recipient_delivery_center": null
},
"additional_services": [
{
    "additional_service_id": 1
}
],
"packages": [
{
    "barcode": "",
    "barcode_type": 1,
    "barcode_client": "test1",
    "weight": 3
},
{
    "barcode": "",
    "barcode_type": 1,
    "barcode_client": "test2",
    "weight": 3
}
]
},
],
"return_address_label": true
}
',
CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUyOjI2NzU5ODQwLCJ1YmY0jE2Njk3M0M1MzQsImV4cCI6MTY2OTcyNzZkNCwiaWF0IjoxNjY5NzEzNTM0fQ.bilmFmhsbCADq8-GK7Apzrf35eyi3AMv0jiMZLLEgCI',

```

```

    'Content-Type: application/json'
  ),
));

$response = curl_exec($curl);

curl_close($curl);
echo $response;

```

## 2.2 CancelShipmentOrders

### C#

```

var client = new RestClient("https://dxwebapit.posta.hr:9020/api/shipment/cancel_shipment_o
rders");
client.Timeout = -1;
var request = new RestRequest(Method.POST);
request.AddHeader("Authorization", "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREU
iOiI2NzU5ODQiLCJmYm9jE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiYW0IjoxNjY5NzE5MjE2fQ.sHoK1KL
YqbM--QjxqDJ_0rcD0njgYUWo4rraBIIdbYM");
request.AddHeader("Content-Type", "application/json");
var body = @"{
" + "\n" +
@"    ""parcels"": [
" + "\n" +
@"      {
" + "\n" +
@"        ""client_reference_number"": ""test1""
" + "\n" +
@"      },
" + "\n" +
@"      {
" + "\n" +
@"        ""client_reference_number"": ""test2""
" + "\n" +
@"      }
" + "\n" +
@"    ]
" + "\n" +
@"}";
request.AddParameter("application/json", body, ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);

```

## JS

```
var settings = {
  "url": "https://dxwebapit.posta.hr:9020/api/shipment/cancel_shipment_orders",
  "method": "POST",
  "timeout": 0,
  "headers": {
    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJ1YmYiOiJlE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiYW0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ_0rcD0njgYUWo4rraBIIdDbyM",
    "Content-Type": "application/json"
  },
  "data": JSON.stringify({
    "parcels": [
      {
        "client_reference_number": "test1"
      },
      {
        "client_reference_number": "test2"
      }
    ]
  }),
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

## PHP – cURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => 'https://dxwebapit.posta.hr:9020/api/shipment/cancel_shipment_orders',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => '',
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => 'POST',
  CURLOPT_POSTFIELDS => '{
    "parcels": [
      {
        "client_reference_number": "test1"
      },
      {
        "client_reference_number": "test2"
      }
    ]
  }'
```

```

}
',
CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQlLCJyYmYiOiJlMjNjMzMTkyMTYsImV4cCI6MTY2OTczMzYxNiwiYWw0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ_0rcD0njgYUWo4rraBIIdByM',
    'Content-Type: application/json'
),
));

$response = curl_exec($curl);

curl_close($curl);
echo $response;

```

## 2.3 GetShipmentStatus

### C#

```

var client = new RestClient("https://dxwebapit.posta.hr:9020/api/shipment/get_shipment_status");
client.Timeout = -1;
var request = new RestRequest(Method.GET);
request.AddHeader("Authorization", "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQlLCJyYmYiOiJlMjNjMzMTkyMTYsImV4cCI6MTY2OTA0MzQzNCwiaWF0IjoxNjY5MDI5MjE2fQ.G1bC0hsBhtoub5uhk-lszDjT3Sb8SF1wmZ3dRLvn1Ho");
request.AddHeader("Content-Type", "application/json");
var body = @"{
  ""barcodes"": [
    {
      ""barcode"": ""LE347087966HR""
    },
    {
      ""barcode"": ""LE347087952HR""
    }
  ]
}";
request.AddParameter("application/json", body, ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);

```

## JS

```
var settings = {
  "url": "https://dxwebapit.posta.hr:9020/api/shipment/get_shipment_status",
  "method": "GET",
  "timeout": 0,
  "headers": {
    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJmYmYiOiJlE2NjkwMjkwMzQsImV4cCI6MTY2OTA0MzQzNCwiaWF0IjoxNjY5MDI5MDM0fQ.G1bC0hsBhtoub5uhk-lszDjT3Sb8SF1wmZ3dRLvnlHo",
    "Content-Type": "application/json"
  },
  "data": JSON.stringify({
    "barcodes": [
      {
        "barcode": "LE347087966HR"
      },
      {
        "barcode": "LE347087952HR"
      }
    ]
  }),
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

## PHP – cURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => 'https://dxwebapit.posta.hr:9020/api/shipment/get_shipment_status',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => '',
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => 'GET',
  CURLOPT_POSTFIELDS => '{ "barcodes": [
    {
      "barcode": "LE347087966HR"
    },
    {
      "barcode": "LE347087952HR"
    }
  ]
}'
);
```

```

',
  CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJ1YmYiOiJlZ2NjkwMjkwMzQsImV4cCI6MTY2OTA0MzQzNCwiaWF0IjoxNjY5MDI5MDM0fQ.G1bC0hsBhtoub5uhk-lszDjT3Sb8SF1wmZ3dRLvnHo',
    'Content-Type: application/json'
  ),
));

$response = curl_exec($curl);
curl_close($curl);
echo $response;

```

## 2.4 GetShippingLabels

### C#

```

var client = new RestClient("https://dxwebapit.posta.hr:9020/api/shipment/get_shipping_labels");
client.Timeout = -1;
var request = new RestRequest(Method.GET);
request.AddHeader("Authorization", "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJ1YmYiOiJlZ2NjkwOTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5MTk0NjE0fQ.CZF54o_51Qq4WT-fQ8jDCDQbNIMZRHe26JBV41s9uyI");
request.AddHeader("Content-Type", "application/json");
var body = @"{
" + "\n" +
@"""client_reference_number"": ""250797204153-R-000124894291002"",
" + "\n" +
@"""barcodes"": [
" + "\n" +
@"    {
" + "\n" +
@"        ""barcode"": ""LE90001297HR""
" + "\n" +
@"    },
" + "\n" +
@"    {
" + "\n" +
@"        ""barcode"": ""LE90001310HR""
" + "\n" +
@"    }
" + "\n" +
@"    ],
" + "\n" +
@"""A4"":false
" + "\n" +
@"}";
request.AddParameter("application/json", body, ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);

```

## JS

```
var settings = {
  "url": "https://dxwebapit.posta.hr:9020/api/shipment/get_shipping_labels",
  "method": "GET",
  "timeout": 0,
  "headers": {
    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJmYmYiOiJlE2NjksOTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5MTk0NjE0fQ.CZF54o_51Qq4WT-fQ8jDCDQbNIMZRHe26JBV41s9uyI",
    "Content-Type": "application/json"
  },
  "data": JSON.stringify({
    "client_reference_number": "250797204153-R-000124894291002",
    "barcodes": [
      {
        "barcode": "LE900001297HR"
      },
      {
        "barcode": "LE900001310HR"
      }
    ],
    "A4": false
  }),
};

$.ajax(settings).done(function (response) {
  console.log(response);
});
```

## PHP – cURL

```
<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => 'https://dxwebapit.posta.hr:9020/api/shipment/get_shipping_labels',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => '',
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => 'GET',
  CURLOPT_POSTFIELDS => '{
"client_reference_number": "250797204153-R-000124894291002",
"barcodes": [
  {
    "barcode": "LE900001297HR"
  },
  {
    "barcode": "LE900001310HR"
  }
]
}'
);
```



```

    }
  ],
  "A4":false
}
',
  CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREU0i2NzU5ODQ0LCJ1YmYiOiJlE2NjY2NjY2OTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5MTk0NjE0fQ.CZF54o_51Qq4WT-fQ8jDCDQbNIMZRHe26JBV41s9uyI',
    'Content-Type: application/json'
  ),
));

$response = curl_exec($curl);

curl_close($curl);
echo $response;

```

### 3.1 GetParcelDeliveryPoint

#### C#

```

var client = new RestClient("https://dxwebapit.posta.hr:9020/api/delivery_point/get_parcel_delivery_point");
client.Timeout = -1;
var request = new RestRequest(Method.GET);
request.AddHeader("Authorization", "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREU0i2NzU5ODQ0LCJ1YmYiOiJlE2NjY2NjY2OTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KL YqbM--QjxqDJ_0rcD0njgYUWo4rraBIIdbyM");
request.AddHeader("Content-Type", "application/json");
var body = @"{
" + "\n" +
@"""facilityType": "PAK""",
" + "\n" +
@"""nextWeek": 0,
" + "\n" +
@"""searchText": "62102""
" + "\n" +
@"}";
request.AddParameter("application/json", body, ParameterType.RequestBody);
IRestResponse response = client.Execute(request);
Console.WriteLine(response.Content);

```

#### JS

```

var settings = {
  "url": "https://dxwebapit.posta.hr:9020/api/delivery_point/get_parcel_delivery_point",
  "method": "GET",
  "timeout": 0,
  "headers": {

```

```

    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJ0YmYiOiJlZ2NjM3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiYWV0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ_0rcD0njgYUWo4rraBIIdByM",
    "Content-Type": "application/json"
  },
  "data": JSON.stringify({
    "facilityType": "PAK",
    "nextWeek": 0,
    "searchText": "62102"
  }),
};

$.ajax(settings).done(function (response) {
  console.log(response);
});

```

## PHP – cURL

```

<?php

$curl = curl_init();

curl_setopt_array($curl, array(
  CURLOPT_URL => 'https://dxwebapit.posta.hr:9020/api/delivery_point/get_parcel_delivery_point',
  CURLOPT_RETURNTRANSFER => true,
  CURLOPT_ENCODING => '',
  CURLOPT_MAXREDIRS => 10,
  CURLOPT_TIMEOUT => 0,
  CURLOPT_FOLLOWLOCATION => true,
  CURLOPT_HTTP_VERSION => CURL_HTTP_VERSION_1_1,
  CURLOPT_CUSTOMREQUEST => 'GET',
  CURLOPT_POSTFIELDS => '{
"facilityType": "PAK",
"nextWeek": 0,
"searchText": "62102"
}',
  CURLOPT_HTTPHEADER => array(
    'Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJ0YmYiOiJlZ2NjM3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiYWV0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ_0rcD0njgYUWo4rraBIIdByM',
    'Content-Type: application/json'
  ),
));

$response = curl_exec($curl);

curl_close($curl);
echo $response;

```