******

***HP Shipping Service API Technical documentation***

Table of contents

[1. Change log 4](#_Toc121466728)

[2. Getting started 5](#_Toc121466729)

[3. Authorization – DX\_Security\_WebAPI 6](#_Toc121466730)

[Response parameters 6](#_Toc121466731)

[4. API URL-s 7](#_Toc121466732)

[5. Shipment methods 8](#_Toc121466733)

[5.1. CreateShipmentOrders 8](#_Toc121466734)

[Definition 8](#_Toc121466735)

[Request parameters 8](#_Toc121466736)

[Request example 11](#_Toc121466737)

[Response example 13](#_Toc121466738)

[Response parameters 14](#_Toc121466739)

[5.2. CancelShipmentOrders 14](#_Toc121466740)

[Definition 14](#_Toc121466741)

[Request parameters 14](#_Toc121466742)

[Request example 15](#_Toc121466743)

[Response example 15](#_Toc121466744)

[Response parameters 15](#_Toc121466745)

[5.3. GetShipmentStatus 16](#_Toc121466746)

[Definition 16](#_Toc121466747)

[Request parameters 16](#_Toc121466748)

[Request example 16](#_Toc121466749)

[Response example 16](#_Toc121466750)

[Response parameters 17](#_Toc121466751)

[5.4. GetShippingLabels 18](#_Toc121466752)

[Definition 18](#_Toc121466753)

[Request parameters 18](#_Toc121466754)

[Request example 18](#_Toc121466755)

[Response example 19](#_Toc121466756)

[Response parameters 19](#_Toc121466757)

[6. Delivery point methods 20](#_Toc121466758)

[6.1. GetParcelDeliveryPoint 20](#_Toc121466759)

[Definition 20](#_Toc121466760)

[Request parameters 20](#_Toc121466761)

[Request example 20](#_Toc121466762)

[Response example 21](#_Toc121466763)

[Appendix A – SCANS - STATUS AND REASONS FOR PARCEL COLLECTION NOT DONE AND NON-DELIVERY 23](#_Toc121466764)

[**STATUS** 23](#_Toc121466765)

[REASONS OF PARCEL COLLECTION IS NOT DONE (NPR) 23](#_Toc121466766)

[REASONS OF NON-DELIVERY (IND) 23](#_Toc121466767)

[Appendix B – List of comments on scan NPA 25](#_Toc121466768)

[Appendix C – Examples of calling methods C# i JS and PHP-cURL 28](#_Toc121466769)

# Change log

|  |  |  |
| --- | --- | --- |
| Change No. | Since | Description |
| 1 | 28.11.2022. | Initial version |

# 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/

# Authorization – DX\_Security\_WebAPI

To access the DXWebAPI service, it is necessary to make an authorization process by calling service:

* **DX\_Security\_WebAPI**

DX\_Security\_WebAPI gives a **bearer token** as a response which must then **be inserted into the *header of the DXWebAPI service call*.**

Bearer token duration is 4 hours.

For calling DX\_Security\_WebAPI service you will need:

* username
* password

provided by Hrvatska pošta d.d..

|  |  |
| --- | --- |
| Method | URL example - TEST |
| POST | https://dxwebapit.posta.hr:9000/api/authentication/client\_auth |

Request example:

{

    "username": "string",

    "password": "string"

}

Response example:

{

    "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkyOTU1ODgsImV4cCI6MTY2OTMwOTk4OCwiaWF0IjoxNjY5Mjk1NTg4fQ.Gq0jwJNDfMH0KOSXpkYGBeMpNILtksVYXtTx4OVwcLw",

 "expiresIn": "14400"

}

### Response parameters

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| accessToken | String | Bearer token |
| expiresIn | Integer | Bearer token duration in **seconds** |

# API URL-s

|  |  |
| --- | --- |
| OKRUŽENJE | URL |
| TEST - DX\_Security\_WebAPI | https://dxwebapit.posta.hr:9000/api/authentication/client\_auth |
| PRODUCTION - DX\_Security\_WebAPI | https://dxwebapi.posta.hr:9000/api/authentication/client\_auth |
| TEST – DXWebAPI | https://dxwebapit.posta.hr:9020/api/{format}/{methodName} |
| PRODUCTION – DXWebAPI | https://dxwebapi.posta.hr:9020/api/{format}/{methodName} |

# Shipment methods

## 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 |
| parcels | List<[ShipmentOrderData](#ShipmentOrderData)> | A list of shipment orders to create | YES |
| return\_address\_label | Bool | Flag for shipping labels in a response for created shipment orders in PDF formatPossible values:**false** – NO**true** – YES |  |

**Class ShipmentOrderData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| client\_reference\_number | String | Unique client ID of shipment order Max. length: 100 | YES |
| service | Integer | Type of servicePossible values:**24** – Pallet shipment**26** – Package 24 D+1**29** – Package 24 D+2**32** – Package 24 D+3**38** – Package 24 D+4 | YES |
| payed\_by | Integer | Shipment payerPossible values:**1** – Sender**2** – RecipientIf NULL Sender is default. |  |
| delivery\_type | Integer | Shipment delivery typePossible values:**1** – Address**2** – Post office**3** – Parcel locker | YES |
| value | Decimal | Insured shipment valueNumber of decimal places: 2 |  |
| payment\_value | Decimal | Cash-on-delivery amountNumber of decimal places: 2 | YES if additional service 9 is chosen (Cash-on-delivery shipment) |
| pickup\_type | Integer | Shipment pickup typePossible values:**1** – Address**2** – Post office**3** – Parcel locker |  |
| parcel\_size | String | Parcel sizePossible values:**X** – Parcel dimension XS**S** – Parcel dimension S**M** – Parcel dimension M**L** – Parcel dimension LParcel dimensions:<https://www.posta.hr/dimenzije-pretinaca/9279> | YES if delivery\_type 3is chosen(Shipment is delivered to a Parcel locker) |
| reference\_field\_B | String | Order reference field BNOT UNIQUE. Max. length: 100 |  |
| reference\_field\_C | String | Order reference field CNOT UNIQUE. Max. length: 100 |  |
| reference\_field\_D | String | Order reference field DNOT UNIQUE. Max. length: 100 |  |
| sender | [SenderData](#SenderData) | Sender's data | YES |
| recipient | [RecipientData](#RecipientData) | Recipient's data | YES |
| additional\_services | List<[AdditionalServiceData](#AdditionalServiceData)> | Additional services |  |
| packages | List<[PackageData](#PackageData)> | Package list | YES |

**SenderData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| sender\_name | String | Sender's name (person or oganization)Max. length: 50 | YES |
| sender\_phone | String | Sender's phone number Max. length: 25 | YES |
| sender\_email | String | Sender's email address Max. length: 25 |  |
| sender\_street | String | Sender's name of the street Max. length: 50 | YES |
| sender\_hnum | String | Sender's number of the house Max. length: 10 | YES |
| sender\_hnum\_suffix | String | Sender's suffix number of the house Max. length: 10 | YES |
| sender\_zip | String | Sender's area zip code Max. length: 5 | YES |
| sender\_city | String | Sender's town or village name Max. length: 25 | YES |
| sender\_pickup\_center | String | Sender's post office code or parcel locker for pickupMax. length: 5 | YES if pickup\_type 2 or 3is chosen |

**RecipientData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| recipient\_name | String | Recipient's name (person or oganization)Max. length: 50 | YES |
| recipient\_phone | String | Recipient's phone number Max. length: 25 | YES |
| recipient\_email | String | Recipient's email address Max. length: 25 |  |
| recipient\_street | String | Recipient's name of the street Max. length: 50 | YES |
| recipient\_hnum | String | Recipient's number of the house Max. length: 10 | YES |
| recipient\_hnum\_suffix | String | Recipient's suffix number of the house Max. length: 10 | YES |
| recipient\_zip | String | Recipient's area zip code Max. length: 5 | YES |
| recipient\_city | String | Recipient's name of the town or village Max. length: 25 | YES |
| recipient\_delivery\_center | String | Recipient's code of post office or parcel locker for deliveryMax. length: 5 | YES if delivery\_type 2 or 3is chosen |

**Class AdditionalServiceData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| additional\_service\_id | Integer | ID of additional servicePossible values:**1** – Personal delivery**3** – Delivery on Saturdays**4** – With the advice of receipt (AR)**9** – Cash-on-delivery shipment**30** – Send a notification to the recipient**32** – Send an e-mail to the recipient**47** – Consolidated shipment |  |

**Class PackageData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| barcode | String | Barcode given by Hrvatska pošta d.d. (before creating shipment order)Max. length: 20 | YES if barcode\_type 0is chosen |
| barcode\_type | Integer | Barcode typePossible values:**0** – Barcode given by Hrvatska pošta d.d. (before creating shipment order)**1** – Barcode given from response after creating shipment order | YES |
| barcode\_client | String | Client's own barcodeMax. length: 500 | YES if barcode\_type 1is chosen |
| weight | Decimal | Package weight in kilogramsNumber 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 |
| ShipmentOrdersList | List<CreateShipmentOrdersResponse> | List of created or rejected shipment orders depending of potentional errors |
| ShipmentsLabel | String | Shipping labels for created shipment orders in PDF format |

**Class CreateShipmentOrdersResponse**

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

## 5.2. CancelShipmentOrders

### Definition

This method is used to cancel shipping orders.

|  |  |
| --- | --- |
| Method | URL example - TEST |
| POST | https://dxwebapit.posta.hr:9020/api/shipment/cancel\_shipment\_orders |

### Request parameters

**Class CancelOrdersData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| parcels | List<[CancelOrderData](#CancelOrderData)> | 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 | **0** – OK**1** – 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 |
| barcodes | List<[BarcodeData](#BarcodeData)> | List of barcodes to retrieve status | YES |

**Class BarcodeData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| barcode | String | Package barcode to retrieve statusMax. 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 |
| Barcode | String | Package barcode  |
| PackageScansList | List<[PackageScanData](#PackageScanData)> | List of package scans |
| ResponseStatus | EnumResponseStatus | **0** – OK**1** – ERROR |
| ErrorCode | Integer | Error code |
| ErrorMessage | String | Error message |

**Class PackageScanData**

|  |  |  |
| --- | --- | --- |
| Parameter | Type | Description |
| Scan | String | Scan code**List of scans:** [Appendix A](#DodatakA) |
| ScanDescription | String | Scan description |
| ScanTime | DateTime | Scan date and time |
| Center | String | Center in which the scan was made |
| Comment | String | Comment on scan |
| CommentNPA | String | Comment on scan NPA**List of comments:** [Appendix B](#DodatakB) |

##

## 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 |
| client\_reference\_number | String | Unique client ID of shipment orderMax. length: 100 | YES if barcodes is NULL or EMPTY |
| barcodes | List<[BarcodeData](#ClassBarcodeData)> | List of barcodes to retrieve labels | YES if CRNis NULL or EMPTY |
| A4 | Bool | Flag for A4 paper size (4 labels on 1 page)Possible values:**false** – NO**true** – YESIf NULL 0 is default. |  |

**Class BarcodeData**

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Type | Description | REQUIRED |
| barcode | String | Package barcodeMax. 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 |
| PackageLabel | String | Package labels of created shipment orders in „base64“ string format |
| ResponseStatus | EnumResponseStatus | **0** – OK**1** – ERROR |
| ErrorCode | Integer | Error code |
| ErrorMessage | String | Error message |

# Delivery point methods

## 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 |
| facilityType | String | Parameter that filters data on post offices and parcel lockers Possible values:**ALL** – Retrieves post offices and parcel lockers **PU** – Retrieves only post lockers **PAK** – Retrieves only parcel lockers  | YES |
| nextWeek | 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 Possible values:**0** – Retrieves post office working hours for the current week **1** – Retrieves post office working hours for the current and the following week  | YES |
| searchText | 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 "postaInfoList" and one for parcel lockers "paketomatInfoList".

PostaInfoList

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 |
| weekDay  | Integer | Brojčana oznaka dana u tjednu 1 – ponedjeljak 2 – utorak . . 7 - nedjelja |
| date | String | Designates the date  |
| openingTime | String | Designates the opening time of the postal office |
| closingTime | String | Designates the closing time of the postal office  |
| Closed | 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 dana 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. **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;

* 1. **CreateShipemntOrders**

**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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTM1MzQsImV4cCI6MTY2OTcyNzkzNCwiaWF0IjoxNjY5NzEzNTM0fQ.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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTM1MzQsImV4cCI6MTY2OTcyNzkzNCwiaWF0IjoxNjY5NzEzNTM0fQ.bilmFmhsbCADq8-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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTM1MzQsImV4cCI6MTY2OTcyNzkzNCwiaWF0IjoxNjY5NzEzNTM0fQ.bilmFmhsbCADq8-GK7Apzrf35eyi3AMv0jiMZLLEgCI',

    'Content-Type: application/json'

  ),

));

$response = curl\_exec($curl);

curl\_close($curl);

**echo** $response;

* 1. **CancelShipmentOrders**

**C#**

**var** client = **new** RestClient("https://dxwebapit.posta.hr:9020/api/shipment/cancel\_shipment\_orders");

client.Timeout = -1;

**var** request = **new** RestRequest(Method.POST);

request.AddHeader("Authorization", "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ\_0rcD0njgYUWo4rraBIdDbyM");

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" +

@"    ]

" + "\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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ\_0rcD0njgYUWo4rraBIdDbyM",

    "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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ\_0rcD0njgYUWo4rraBIdDbyM',

    'Content-Type: application/json'

  ),

));

$response = curl\_exec($curl);

curl\_close($curl);

**echo** $response;

* 1. **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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkwMjkwMzQsImV4cCI6MTY2OTA0MzQzNCwiaWF0IjoxNjY5MDI5MDM0fQ.G1bC0hsBhtoub5uhk-lszDjT3Sb8SF1wmZ3dRLvnlHo");

request.AddHeader("Content-Type", "application/json");

**var** body = @"{    ""barcodes"": [

" + "\n" +

@"        {

" + "\n" +

@"            ""barcode"": ""LE347087966HR""

" + "\n" +

@"        },

" + "\n" +

@"        {

" + "\n" +

@"            ""barcode"": ""LE347087952HR""

" + "\n" +

@"        }

" + "\n" +

@"

" + "\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/get\_shipment\_status",

  "method": "GET",

  "timeout": 0,

  "headers": {

    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkwMjkwMzQsImV4cCI6MTY2OTA0MzQzNCwiaWF0IjoxNjY5MDI5MDM0fQ.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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkwMjkwMzQsImV4cCI6MTY2OTA0MzQzNCwiaWF0IjoxNjY5MDI5MDM0fQ.G1bC0hsBhtoub5uhk-lszDjT3Sb8SF1wmZ3dRLvnlHo',

    'Content-Type: application/json'

  ),

));

$response = curl\_exec($curl);

curl\_close($curl);

**echo** $response;

* 1. **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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkxOTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5MTk0NjE0fQ.CZF54o\_51Qq4WT-fQ8jDCDQbNIMZRHe26JBV41s9uyI");

request.AddHeader("Content-Type", "application/json");

**var** body = @"{

" + "\n" +

@"""client\_reference\_number"": ""250797204153-R-000124894291002"",

" + "\n" +

@"""barcodes"": [

" + "\n" +

@"        {

" + "\n" +

@"            ""barcode"": ""LE900001297HR""

" + "\n" +

@"        },

" + "\n" +

@"        {

" + "\n" +

@"            ""barcode"": ""LE900001310HR""

" + "\n" +

@"        }

" + "\n" +

@"    ],

" + "\n" +

@"""A4"":false

" + "\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/get\_shipping\_labels",

  "method": "GET",

  "timeout": 0,

  "headers": {

    "Authorization": "Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkxOTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5MTk0NjE0fQ.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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2NjkxOTQ2MTQsImV4cCI6MTY2OTIwOTAxNCwiaWF0IjoxNjY5MTk0NjE0fQ.CZF54o\_51Qq4WT-fQ8jDCDQbNIMZRHe26JBV41s9uyI',

    'Content-Type: application/json'

  ),

));

$response = curl\_exec($curl);

curl\_close($curl);

**echo** $response;

* 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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ\_0rcD0njgYUWo4rraBIdDbyM");

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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ\_0rcD0njgYUWo4rraBIdDbyM",

    "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.eyJDRUNPREUiOiI2NzU5ODQiLCJuYmYiOjE2Njk3MTkyMTYsImV4cCI6MTY2OTczMzYxNiwiaWF0IjoxNjY5NzE5MjE2fQ.sHoK1KLYqbM--QjxqDJ\_0rcD0njgYUWo4rraBIdDbyM',

    'Content-Type: application/json'

  ),

));

$response = curl\_exec($curl);

curl\_close($curl);

**echo** $response;