

Manual validation procedure of generated labels in the ORLEN Paczka system

ORLEN Paczka, Warsaw, 04.03.2025

Table of contents

1. Label	3
Label design	3
Required fields to be embedded on the label:	3
Label size	5
Description of the fields on the label	5
2. Webservice	6
Address	6
Methods	6
GiveMeAllLocationWithAllDataWithZipCode	6
GenerateLabelBusinessPack	8

1. Label

Label design

1 Oddz. Dor. WS	2  Kioski RUCHu Automaty Paczkowe Stacje ORLEN Punkty Partnerskie	3 M. Rejon WSA5	
4 Obszar WWW	5 Numer paczki: 3400940212940940	6 resort 02	7 Czas SW
8 			
9 YP POK PPP	10 POK 100001	11 Jmót POK: AB-32	12 Gabaryt M
13 Marszałka Józefa Piłsudskiego skrzyżowanie z Emilii Plater 77 23-465 Przedmieście Szczepieszyńskie			14 Nr Zam. PL100404676/120W 607R6855021447
15 Nadawca	16 S.A Send AmipolSend 17A Send Warszawa Send 11-111		17 adanie 24,99 PLN
			18 Jwaga Pobranie
			19 JWAGA ZWROT
20 Odbiorca	21 usz Kowalski wiejska 4 01-123 Warszawa		22 
23 NAJBLIŻSZE PUNNKTY ODBIORU: Punkt Asd_1 123 0 M Punkt Asd_2 123 1 M Punkt Asd_3 123 2 M			
24  Zawsze blisko, Zawsze wygodnie			

Required fields to be embedded on the label:

1. Delivery branch

- Data for this field is retrieved from the first two characters of <DestinationCode> (GiveMeAllLocationWithAllDataWithZipCode method).

3. Courier microdistrict

- Data for this field is retrieved from <Mikrorejon> (GiveMeAllLocationWithAllDataWithZipCode method).

4. Courier area

- Data for this field is retrieved from <Obszar> (GiveMeAllLocationWithAllDataWithZipCode method).

5. Package number

- The reserved package number for which notification data is transmitted using the PutCustomerPack method.

6. Presort

- Data for this field is retrieved from <Presort> (GiveMeAllLocationWithAllDataWithZipCode method). If the parameter is missing in the API response for GiveMeAllLocationWithAllDataWithZipCode, the field can be omitted.

7. Time. Parameter for courier needs, regarding the availability time of the point for courier deliveries

- Data for this field is retrieved from <Czas> (GiveMeAllLocationWithAllDataWithZipCode method). If the parameter is missing in the API response for GiveMeAllLocationWithAllDataWithZipCode, the field can be omitted.

8. Barcode

- Barcode in Code 128 format. There are no strict guidelines regarding its size. In the case of sending the first label, its readability by scanners is verified by the logistics department.

9. Type of collection point

- Data for this field is retrieved from <PointType> (GiveMeAllLocationWithAllDataWithZipCode method).

10. Unique 6-digit collection point number

- Data for this field is retrieved from <PSD> (GiveMeAllLocationWithAllDataWithZipCode method).

11. Collection point abbreviation

- Data for this field is retrieved from <Skrotnrpok> (GiveMeAllLocationWithAllDataWithZipCode method).

13. Collection point address

- The data in this field should consist of three elements retrieved using the GiveMeAllRUCHZipcode method: <StreetName>, <Zipcode>, <City>.

24. ORLEN Paczki logo

- Mandatory field containing the ORLEN Paczki logo (<https://www.orientparcels.pl/>).

Label size

Height	14 cm
Width	10 cm

Description of the fields on the label

No.	Name	Description
1	<i>Oddz. Dor.</i>	Delivery branch
2	<i>Logo</i>	Logo displayed depending on the partner
3	<i>M. Rejon</i>	Courier micro-zone
4	<i>Obszar</i>	Delivery zone
5	<i>Numer paczki</i>	Parcel number
6	<i>Presort</i>	
7	<i>Czas</i>	Courier-specific parameter regarding the availability time of the pickup point for courier deliveries
8	<i>Kod kreskowy</i>	EAN-13 number of the parcel
9	<i>TYP POK</i>	Pickup point type
10	<i>POK</i>	Unique 6-digit pickup point number
11	<i>Skrót POK</i>	Pickup point abbreviation
12	<i>Gabaryt</i>	Size of the parcel (S, M, L)
13	<i>Adres POK</i>	Pickup point address
14	<i>Nr Zam.</i>	Order number
15	<i>Nadawca</i>	Sender name, sender surname, sender company name
16	<i>Adres nadawcy</i>	Sender street name, building number, postal code, city
17	<i>Nadanie</i>	The shipment fee amount, information about payment for the parcel
18	<i>Uwaga Pobranie</i>	Cash on delivery parcel
19	<i>UWAGA ZWROT</i>	Parcel return
20	<i>Odbiorca</i>	Recipient name, recipient surname, recipient company name
21	<i>Adres odbiorcy</i>	Recipient street name, building number, postal code, city
22	<i>Kod QR</i>	Parcel number as a QR code
23	<i>Najbliższe Punkty Odbioru</i>	The nearest alternative pickup points to which the parcel will be redirected in the event of a malfunction or overcapacity of the originally chosen pickup point
24	<i>Reklama</i>	Advertisement

2. Webservice

Address

Environment	Address
Production	https://api.orientapaczka.pl/WebServicePwRProd/WebServicePwR.asmx
Test	https://api-test.orientapaczka.pl/WebServicePwR/WebServicePwR.asmx

Methods

GiveMeAllLocationWithAllDataWithZipCode

The method returns extended data for all available points within the ORLEN Paczka service. Additionally, for each of the points, three alternative points located closest to the specified pickup point are returned.

Sample request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GiveMeAllLocationWithAllDataWithZipCode xmlns="https://91.242.220.103/WebServicePwR" />
  </soap:Body>
</soap:Envelope>
```

Sample response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GiveMeAllLocationWithAllDataWithZipCodeResponse xmlns="https://91.242.220.103/WebServicePwR">
      <GiveMeAllLocationWithAllDataWithZipCodeResult>
        <xs:schema id="NewDataSet" xmlns="" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:UseCurrentLocale="true">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="LocationWithAllData2">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="DestinationCode" type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:element>
        </xs:schema>
      </GiveMeAllLocationWithAllDataWithZipCodeResult>
    </GiveMeAllLocationWithAllDataWithZipCodeResponse>
  </soap:Body>
</soap:Envelope>
```

```

<xs:element name="StreetName" type="xs:string" minOccurs="0" />
<xs:element name="BuildingNumber" type="xs:string" minOccurs="0" />
<xs:element name="City" type="xs:string" minOccurs="0" />
<xs:element name="Zipcode" type="xs:string" minOccurs="0" />
<xs:element name="District" type="xs:string" minOccurs="0" />
<xs:element name="Longitude" type="xs:string" minOccurs="0" />
<xs:element name="Latitude" type="xs:string" minOccurs="0" />
<xs:element name="Province" type="xs:string" minOccurs="0" />
<xs:element name="CashOnDelivery" type="xs:boolean" minOccurs="0" />
<xs:element name="OpeningHours" type="xs:string" minOccurs="0" />
<xs:element name="Location" type="xs:string" minOccurs="0" />
<xs:element name="PSD" type="xs:string" minOccurs="0" />
<xs:element name="Available" type="xs:string" minOccurs="0" />
<xs:element name="Obszar" type="xs:string" minOccurs="0" />
<xs:element name="Mikrorejon" type="xs:string" minOccurs="0" />
<xs:element name="Skrotnrpok" type="xs:string" minOccurs="0" />
<xs:element name="Sortownia" type="xs:string" minOccurs="0" />
<xs:element name="Presort" type="xs:string" minOccurs="0" />
<xs:element name="Czas" type="xs:string" minOccurs="0" />
<xs:element name="PointType" type="xs:string" minOccurs="0" />
<xs:element name="NearestPoints"
msdata:DataType="System.Collections.Generic.List`1[[Pwr.Common.Models.NearestPoint,
Pwr.Common.Models, Version=1.0.0.0, Culture=neutral, PublicKeyToken=null]], mscorlib,
Version=4.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089" type="xs:anyType"
minOccurs="0" />
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:schema>
<diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">
<NewDataSet xmlns="">
<LocationWithAllData2 diffgr:id="LocationWithAllData21" msdata:rowOrder="0">
  <DestinationCode>KL-895926-J2-55</DestinationCode>
  <StreetName>CHROBREGO</StreetName>
  <BuildingNumber>6</BuildingNumber>
  <City>Jędrzejów</City>
  <Zipcode>28-300</Zipcode>
  <District>Jędrzejów</District>
  <Longitude>20.2861596</Longitude>
  <Latitude>50.6426837</Latitude>
  <Province>świętokrzyskie</Province>
  <CashOnDelivery>>false</CashOnDelivery>
  <OpeningHours>Pn-Pt:00:00-24:00, So:00:00-24:00, Nd:00:00-24:00</OpeningHours>
  <Location>Automat na terenie biuro ubezpieczeniowe</Location>
  <PSD>895926</PSD>
  <Available>T</Available>
  <Obszar />

```

```

<Mikrorejon>KLJ2</Mikrorejon>
<Skrotnrpok>J2-55</Skrotnrpok>
<Sortownia />
<Presort>06</Presort>
<Czas>JJ</Czas>
<PointType>APM</PointType>
<NearestPoints>
  <NearestPoint>
    <Name>Jędrzejów Przytkowskiego 41A/1</Name>
    <Distance>142</Distance>
  </NearestPoint>
  <NearestPoint>
    <Name>Jędrzejów Armii Krajowej 1A</Name>
    <Distance>506</Distance>
  </NearestPoint>
  <NearestPoint>
    <Name>Jędrzejów al. Piłsudskiego 4</Name>
    <Distance>542</Distance>
  </NearestPoint>
</NearestPoints>
</LocationWithAllData2>

```

Returned parameters

<DestinationCode>	POPer code	15 char
<StreetName>	street from the recipient's address	30 char
<City>	city from the recipient's address	20 char
<Zipcode>	zipcode	6 char
<District>	district	20 char
<Longitude>	longitude	10 char
<Latitude>	latitude	10 char
<Province>	province	20 char
<CashOnDelivery>	is Cash On Delivery service available at the pickup point	(true/false) boolean
<OpeningHours>	pickup point opening hours	50 char
<Location>	description of the pickup point location	255 char
<PSD>	unique 6-digit pickup point ID number	6 char
<Available>	is the point available in the ORLEN Paczka service	(T/N) 1 char
<Obszar>	pick-up point area	20 char
<Mikrorejon>	microregion of the pickup point	20 char
<Skrotnrpok>	pickup point ID abbreviation	20 char
<Sortownia>	the sorting facility assigned to the pickup point	20 char
<Presort>		20 char
<Czas>	the parameter for courier purposes, regarding the availability time of the pickup point for courier deliveries	20 char
<PointType>	pickup point type	3 char
<NearestPoints>	a list of the three closest alternative pick-up points, providing: point description <Name> and distance from the specified pick-up point <Distance>	list of points (1...n)

GenerateLabelBusinessPack

This method is used to notify the ORLEN Paczka parcel, in response it returns a PDF label.

Sample request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GenerateLabelBusinessPack xmlns="https://91.242.220.103/WebServicePwR">
      <PartnerID>PWR0000000</PartnerID>
      <PartnerKey>XXXXXXXXXX</PartnerKey>
      <DestinationCode>WS-123456-00-00</DestinationCode>
      <AlternativeDestinationCode></AlternativeDestinationCode>
      <BoxSize></BoxSize>
      <PackValue></PackValue>
      <CashOnDelivery>N</CashOnDelivery>
      <AmountCashOnDelivery></AmountCashOnDelivery>
      <Insurance>N</Insurance>
      <EMail>test@wp.pl</EMail>
      <FirstName>Testowelmie</FirstName>
      <LastName>TestoweNazwisko</LastName>
      <CompanyName></CompanyName>
      <StreetName>Annopol</StreetName>
      <BuildingNumber>17</BuildingNumber>
      <FlatNumber></FlatNumber>
      <City>Warszawa</City>
      <PostCode>00-123</PostCode>
      <PhoneNumber>123456789</PhoneNumber>
      <SenderEMail>testyruch@gmail.com</SenderEMail>
      <SenderFirstName></SenderFirstName>
      <SenderLastName></SenderLastName>
      <SenderCompanyName>SenderCompany</SenderCompanyName>
      <SenderStreetName>Kwiatowa</SenderStreetName>
      <SenderBuildingNumber>11</SenderBuildingNumber>
      <SenderFlatNumber>22c</SenderFlatNumber>
      <SenderCity>Krakow</SenderCity>
      <SenderPostCode>00-321</SenderPostCode>
      <SenderPhoneNumber>999888777</SenderPhoneNumber>
      <SenderOrders></SenderOrders>
      <ReturnDestinationCode></ReturnDestinationCode>
      <ReturnEMail></ReturnEMail>
      <ReturnFirstName></ReturnFirstName>
      <ReturnLastName></ReturnLastName>
      <ReturnCompanyName></ReturnCompanyName>
      <ReturnStreetName></ReturnStreetName>
      <ReturnBuildingNumber></ReturnBuildingNumber>
      <ReturnFlatNumber></ReturnFlatNumber>
      <ReturnCity></ReturnCity>
      <ReturnPostCode></ReturnPostCode>
      <ReturnPhoneNumber></ReturnPhoneNumber>
      <ReturnPack>N</ReturnPack>
      <TransferDescription></TransferDescription>
      <PrintAdress>1</PrintAdress>
```

```

    <ReturnAvailable>N</ReturnAvailable>
    <ReturnQuantity></ReturnQuantity>
    <PrintType>1</PrintType>
  </GenerateLabelBusinessPack>
</soap:Body>
</soap:Envelope>

```

Sample response

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GenerateLabelBusinessPackResponse xmlns="https://91.242.220.103/WebServicePwR">
      <GenerateLabelBusinessPackResult>
        <xs:schema id="NewDataSet" xmlns="" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
          <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:UseCurrentLocale="true">
            <xs:complexType>
              <xs:choice minOccurs="0" maxOccurs="unbounded">
                <xs:element name="GenerateLabelBusinessPack">
                  <xs:complexType>
                    <xs:sequence>
                      <xs:element name="Err" type="xs:string" minOccurs="0" />
                      <xs:element name="ErrDes" type="xs:string" minOccurs="0" />
                      <xs:element name="PackCode_RUCH" type="xs:string" minOccurs="0" />
                      <xs:element name="DestinationCode" type="xs:string" minOccurs="0" />
                      <xs:element name="DestinationId" type="xs:int" minOccurs="0" />
                      <xs:element name="PackPrice" type="xs:int" minOccurs="0" />
                      <xs:element name="PackPaid" type="xs:boolean" minOccurs="0" />
                      <xs:element name="ReturnDestinationId" type="xs:int" minOccurs="0" />
                      <xs:element name="ReturnDestinationCode" type="xs:string" minOccurs="0" />
                      <xs:element name="NameCL" type="xs:string" minOccurs="0" />
                      <xs:element name="NrCL" type="xs:string" minOccurs="0" />
                    </xs:sequence>
                  </xs:complexType>
                </xs:element>
              </xs:choice>
            </xs:complexType>
          </xs:schema>
          <diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">
            <NewDataSet xmlns="">
              <GenerateLabelBusinessPack diffgr:id="GenerateLabelBusinessPack1" msdata:rowOrder="0">
                <Err>000</Err>
                <ErrDes>saved</ErrDes>
                <PackCode_RUCH>2000000000000</PackCode_RUCH>
                <DestinationCode>WS-12345-00-00</DestinationCode>
                <DestinationId>12345</DestinationId>
                <PackPrice>1099</PackPrice>
                <PackPaid>true</PackPaid>
              </GenerateLabelBusinessPack>
            </NewDataSet>
          </diffgr:diffgram>
        </xs:schema>
      </GenerateLabelBusinessPackResult>
    </GenerateLabelBusinessPackResponse>
  </soap:Body>
</soap:Envelope>

```

```

        <NameCL>BYDGOSZCZ</NameCL>
        <NrCL>0902</NrCL>
    </GenerateLabelBusinessPack>
</NewDataSet>
</diffgr:diffgram>
</GenerateLabelBusinessPackResult>
<LabelData>{...}</LabelData>
</GenerateLabelBusinessPackResponse>
</soap:Body>
</soap:Envelope>

```

Requests' parameters

<PartnerID>	customer's ID given by RUCH	(REQUIRED) 10 char
<PartnerKey>	customer's key given by RUCH	(REQUIRED) 10 char
<DestinationCode>	POPeR code	(REQUIRED) 15 char
<AlternativeDestinationCode>	alternative POPeR code	15 char
<BoxSize>	parcel size	(REQUIRED) 10 char
<PackValue>	parcel value (in grosze)	6 char
<CashOnDelivery>	is the Cash On Delivery service available?	(true/false) boolean
<AmountCashOnDelivery>	Cash On Delivery amount (in grosze)	6 char
<Insurance>	is the parcel insured?	(true/false) boolean
<EMail>	recipient's e-mail address	60 char
<FirstName>	recipient's name	30 char
<LastName>	recipient's surname	30 char
<CompanyName>	recipient's company name	70 char
<StreetName>	recipient's street name	30 char
<BuildingNumber>	recipient's building number	10 char
<FlatNumber>	recipient's flat number	10 char
<City>	recipient's city	30 char
<PostCode>	recipient's postcode	6 char
<PhoneNumber>	sender's phone number	(REQUIRED) 9 char
<SenderEMail>	sender's e-mail address	(REQUIRED) 60 char
<SenderFirstName>	sender's name	(REQUIRED) 30 char
<SenderLastName>	sender's surname	(REQUIRED) 30 char
<SenderCompanyName>	sender's company name	70 char
<SenderStreetName>	sender's street name	(REQUIRED) 30 char
<SenderBuildingNumber>	sender's building number	(REQUIRED) 10 char
<SenderFlatNumber>	sender's flat number	10 char
<SenderCity>	sender's city	(REQUIRED) 30 char
<SenderPostCode>	sender's postcode	(REQUIRED) 6 char
<SenderPhoneNumber>	sender's phone number	(REQUIRED) 9 char
<SenderOrders>	sender's order number	30 char
<ReturnDestinationCode>	Return POPeR code	15 char
<ReturnEMail >	return e-mail address	60 char
<ReturnFirstName>	person's name for return	30 char
<ReturnLastName>	person's surname for return	30 char
<ReturnCompanyName>	company name for return	70 char
<ReturnStreetName>	street name for return	30 char
<ReturnBuildingNumber>	building number for return	10 char
<ReturnFlatNumber>	flat number for return	10 char
<ReturnCity>	city for return	30 char

<ReturnPostCode>	postcode for return	6 char
<ReturnPhoneNumber>	phone number for return	9 char
<ReturnPack>	return parcel (T – if return)	1 char
<TransferDescription>	transfer title for Cash On Delivery parcels	70 char
<PrintAdress>	selection of address for printout in case of return (1 – dispatch address, 2 – return address)	(REQUIRED) 1 char
<ReturnAvailable>	ability to send a return parcel (T - if yes)	1 char
<ReturnQuantity>	number of return parcels	1 char
<PrintType>	choice of label for printout (1 – full address, 2 – anonymous address)	(REQUIRED) 1 char