

Dokumentacja podłączeniowa dla procesu przenoszenia danych osobowych

Czyli opis jak skorzystać z usługi: **rodoTransferService**

Spis treści

| | |
|--|---|
| Dokumentacja podłączeniowa dla procesu przenoszenia danych osobowych | 1 |
| Przed rozpoczęciem..... | 3 |
| Opis interfejsu do transferu danych..... | 4 |
| Wywołanie..... | 5 |
| Odpowiedź..... | 6 |
| Dodatek A – Pliki z definicją interfejsu | 7 |

Przed rozpoczęciem

Aby przesłać lub otrzymywać dane osobowe do firmy P4 należy przede wszystkim zgłosić się do działu technicznego w celu podpisania własnego certyfikatu przez Play CA, co sprawi, że będziecie Państwo traktowani przez system do przenoszenia numerów jako podmiot uprawniony do przesyłania danych. Certyfikat powinien dotyczyć domeny do której państwo macie dostęp i możecie wykonywać na niej zmiany. Kontakt można nawiązać pisząc na skrzynkę _nomad@play.pl.

Gdy już państwo uzyskacie podpisany przez nas certyfikat, będziecie mogli przestąpić do automatycznego przekazywania lub odbierania od nas danych osobowych zgodnie z poniższymi wytycznymi.

Opis interfejsu do transferu danych

Interfejs przesyłania danych osobowych jest zrealizowany w formie Web Service'u SOAPowego, dla którego pliki z definicją są dołączone na samym końcu w załączniku A.

Adres interfejsu to: https://nomad.playmobile.pl/data_migrate

Interfejs posiada tylko jedną operację - przesłanie danych osobowych do innego operatora, zwaną dalej transferem.

Wywołanie

Aby przekazać dane osobowe, należy uzupełnić wywołanie w następujący sposób:

Element `transferDataId` określa id transferu po stronie operatora, który **wysyła** dane. Powinno ono unikalnie identyfikować chęć przekazania danych przez podmiot do innego administratora danych osobowych. W przypadku otrzymania transferu przez **odbiorcę** z wartością `transferDataId`, która została już wcześniej przez niego przetworzona, odbiorca powinien zignorować taki request i odpowiedzieć z sukcesem.

Atrybut `transferDataRequestId` określa id konkretnego requestu w ramach transferu i jest używany w celach diagnostycznych. Powinno zawierać losowo wygenerowaną wartość dla każdej próby transferu.

Element `transferDataSet` zawiera przekazywane dane osobowe podmiotu. Jest to zbiór zestawów danych osobowych. Każdy zestaw danych osobowych jest obowiązkowo opisany przez rodzaj - atrybut `transferDataSetType` (*basic, customer, employee, contractor, candidate, other*) określający, czego dotyczy. Zestaw danych osobowych jest luźną strukturą zawierającą przesyłane dane osobowe. W miarę możliwości operator, który wysyła dane powinien uzupełnić **wszystkie** przetwarzane przez niego dane osobowe.

Odpowiedź

W odpowiedzi na transfer danych osobowych należy przekazać te same wartości **transferDataId** oraz **transferDataRequestId**. W przypadku pomyślnego odebrania danych należy wypełnić element *code* wartością 0. W przypadku błędu wartość pola *code* powinna być różna od 0. Element **message** jest opcjonalnym polem zawierającym opis błędu.

Więcej szczegółów jest zawartych w dodatku A z definicją interfejsu.

Dodatek A – Pliki z definicją interfejsu

Plik WSDL

```
<wsdl:definitions name="rodoDataTransferService"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
    xmlns:tns="http://core.proxy.nomad.p4.touk.pl/transfer-ws"
    targetNamespace="http://core.proxy.nomad.p4.touk.pl/transfer-
ws">
  <wsdl:types>
    <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
      <xsd:import namespace="http://core.proxy.nomad.p4.touk.pl/transfer-
ws"
        schemaLocation="rodoDataTransferService.xsd"/>
    </xsd:schema>
  </wsdl:types>
  <wsdl:message name="transferDataRequest">
    <wsdl:part name="parameters" element="tns:transferDataRequest"/>
  </wsdl:message>
  <wsdl:message name="transferDataResponse">
    <wsdl:part name="parameters" element="tns:transferDataResponse"/>
  </wsdl:message>
  <wsdl:message name="genericServiceFault">
    <wsdl:part name="parameters" element="tns:genericServiceFault"/>
  </wsdl:message>
  <wsdl:portType name="nomadTransferPort">
    <wsdl:operation name="transferData">
      <wsdl:documentation>
        Zleca transfer danych osobowych do nowego kontrolera
        (administratora)
      </wsdl:documentation>
      <wsdl:input message="tns:transferDataRequest"/>
      <wsdl:output message="tns:transferDataResponse"/>
      <wsdl:fault name="fault" message="tns:genericServiceFault"/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="nomadTransferBinding" type="tns:nomadTransferPort">
    <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="transferData">
      <soap:operation
soapAction="http://core.proxy.nomad.p4.touk.pl/transfer-ws/transferData"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
      <wsdl:fault name="fault">
        <soap:fault name="fault" use="literal"/>
      </wsdl:fault>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:service name="nomadTransferService">
    <wsdl:port name="nomadTransferPort" binding="tns:nomadTransferBinding">
      <soap:address location="http://core.proxy.nomad.p4.touk.pl/transfer-
ws"/>
    </wsdl:port>
  </wsdl:service>
</wsdl:definitions>
```

Plik rodoDataTransferService.xsd

```

<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://core.proxy.nomad.p4.touk.pl/transfer-ws"
  targetNamespace="http://core.proxy.nomad.p4.touk.pl/transfer-ws">

  <xsd:element name="genericServiceFault">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Odpowiedź informująca o błędzie w trakcie wykonania procedury.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="message" type="xsd:string"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="transferDataRequest">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Typ definiujący zlecenie przeniesienia danych osobowych do nowego
        kontrolera (administratora)
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="transferDataId" type="xsd:string"/>
        <xsd:element name="transferDataSet" type="tns:TransferDataSet"
minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="transferDataRequestId" type="xsd:string"
use="required"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="transferDataResponse">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Typ definiujący odpowiedź na zlecenie przeniesienia danych
        osobowych do nowego kontrolera
        (administratora)
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="transferDataId" type="xsd:string"/>
      </xsd:sequence>
      <xsd:attribute name="transferDataRequestId" type="xsd:string"
use="required"/>
    </xsd:complexType>
  </xsd:element>

  <!-- pozostałe typy danych -->

  <xsd:complexType name="TransferDataSet">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Typ definiujący zestaw danych osobowych (dataset). Rodzaj zestawu
        określony jest w atrybucie: transferDataSetType
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="personNames" type="xsd:string" minOccurs="0"/>
      <xsd:element name="personLastNames" type="xsd:string" minOccurs="0"/>
      <xsd:element name="companyName" type="xsd:string" minOccurs="0"/>
      <xsd:element name="pesel" type="tns:PeselType" minOccurs="0"/>
      <xsd:element name="nip" type="tns:NipType" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

```



```

        <xsd:element name="regon" type="tns:RegonType" minOccurs="0"/>
        <xsd:element name="documentSet" type="tns:DocumentSetType"
minOccurs="0"/>
        <xsd:element name="emailSet" type="tns:EmailSetType" minOccurs="0"/>
        <xsd:element name="phoneNumberSet" type="tns:PhoneNumberSetType"
minOccurs="0"/>
        <xsd:element name="addressSet" type="tns:addressSetType"
minOccurs="0"/>
        <xsd:element name="serviceMsisdnSet" type="tns:PhoneNumberSetType"
minOccurs="0"/>
        <xsd:element name="motherNames" type="xsd:string" minOccurs="0"/>
        <xsd:element name="motherFamilyNames" type="xsd:string"
minOccurs="0"/>
        <xsd:element name="fatherNames" type="xsd:string" minOccurs="0"/>
        <xsd:element name="fatherFamilyNames" type="xsd:string"
minOccurs="0"/>
        <xsd:element name="birthDate" type="xsd:date" minOccurs="0"/>
        <xsd:element name="birthPlace" type="xsd:string" minOccurs="0"/>
        <xsd:element name="birthCountry" type="xsd:string" minOccurs="0"/>
        <xsd:element name="personNationality" type="xsd:string"
minOccurs="0"/>
        <xsd:element name="personGender" type="xsd:string" minOccurs="0"/>
        <xsd:element name="lastEmploymentPosition" type="xsd:string"
minOccurs="0"/>
        <xsd:element name="lastEmploymentStartDate" type="xsd:date"
minOccurs="0"/>
        <xsd:element name="lastEmploymentEndDate" type="xsd:date"
minOccurs="0"/>
        </xsd:sequence>
        <xsd:attribute name="transferDataSetType" type="tns:TransferDataSetType"
use="required"/>
    </xsd:complexType>

    <xsd:complexType name="DocumentSetType">
        <xsd:annotation>
            <xsd:documentation xml:lang="pl">
                Typ definiujący kolekcję dokumentów danej osoby.
            </xsd:documentation>
        </xsd:annotation>
        <xsd:sequence>
            <xsd:element name="document" type="tns:DocumentType"
maxOccurs="unbounded"/>
        </xsd:sequence>
    </xsd:complexType>

    <xsd:complexType name="DocumentType">
        <xsd:annotation>
            <xsd:documentation xml:lang="pl">
                Typ definiujący dokument tożsamości (np. dowód osobisty,
                paszport, prawo jazdy itp). Możliwe atrybuty:
                [type] - typ dokumentu
                [number] - seria/numer dokumentu
                [expirationDate] - data ważności
                [issueDate] - data wystawienia
            </xsd:documentation>
        </xsd:annotation>
        <xsd:attribute name="type" type="xsd:string" use="required"/>
        <xsd:attribute name="number" type="xsd:string" use="required"/>
        <xsd:attribute name="expirationDate" type="xsd:date"/>
        <xsd:attribute name="issueDate" type="xsd:date"/>
    </xsd:complexType>

    <xsd:complexType name="addressSetType">
        <xsd:annotation>
            <xsd:documentation xml:lang="pl">
                Typ definiujący listę adresów danej osoby.
            </xsd:documentation>
        </xsd:annotation>
    </xsd:complexType>

```

```

    <xsd:sequence>
      <xsd:element name="address" type="tns:AddressType"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="AddressType">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Type definiujący adres. Możliwe pola:
        [type] - rodzaj adresu (np. adres zamieszkania, korespondencyjny,
zameldowania itp)
        [street] - nazwa ulicy
        [buildingNumber] - numer budynku
        [apartmentNumber] - numer mieszkania / lokalu w budynku
        [zip] - kod pocztowy
        [city] - miasto / miejscowość
        [district] - dzielnica / gmina
        [province] - prowincja / województwo
        [country] - kraj
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="type" type="tns:AddressTypeType"/>
      <xsd:element name="street" type="xsd:string" minOccurs="0"/>
      <xsd:element name="buildingNumber" type="xsd:string" minOccurs="0"/>
      <xsd:element name="apartmentNumber" type="xsd:string" minOccurs="0"/>
      <xsd:element name="zip" type="xsd:string" minOccurs="0"/>
      <xsd:element name="city" type="xsd:string" minOccurs="0"/>
      <xsd:element name="district" type="xsd:string" minOccurs="0"/>
      <xsd:element name="province" type="xsd:string" minOccurs="0"/>
      <xsd:element name="country" type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:simpleType name="PeselType">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Typ definiujący PESEL.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9]{11,11}"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="NipType">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Typ definiujący NIP.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9]{10}"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="RegonType">
    <xsd:annotation>
      <xsd:documentation xml:lang="pl">
        Typ definiujący REGON.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="[0-9]{9,9}|[0-9]{14,14}"/>
    </xsd:restriction>
  </xsd:simpleType>

```

```

<xsd:complexType name="EmailSetType">
  <xsd:annotation>
    <xsd:documentation xml:lang="pl">
      Typ definiujący kolekcję adresów EMAIL.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="email" type="tns:EmailType"
maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="EmailType">
  <xsd:annotation>
    <xsd:documentation xml:lang="pl">
      Typ definiujący EMAIL (uproszczona, niepełna walidacja).
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,8}"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="PhoneNumberSetType">
  <xsd:annotation>
    <xsd:documentation xml:lang="pl">
      Typ definiujący kolekcję numerów telefonów.
    </xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="phoneNumber" type="tns:PhoneNumberType"
maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:simpleType name="PhoneNumberType">
  <xsd:annotation>
    <xsd:documentation xml:lang="pl">
      Typ definiujący numer telefonu (9 cyfr, bez znaków specjalnych).
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="[0-9]{9}"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="TransferDataSetType">
  <xsd:annotation>
    <xsd:documentation xml:lang="pl">
      Typ definiujący rodzaj zestawu danych osobowych. Możliwe
wartości:
      [basic] - podstawowe dane osobowe
      [customer] - dane osobowe klienta
      [employee] - dane osobowe pracownika
      [contractor] - dane osobowe kontrahenta
      [candidate] - dane osobowe kandydata na pracownika
      [other] - inny zestaw danych osobowych
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="basic"/>
    <xsd:enumeration value="customer"/>
    <xsd:enumeration value="employee"/>
    <xsd:enumeration value="contractor"/>
    <xsd:enumeration value="candidate"/>
    <xsd:enumeration value="other"/>
  </xsd:restriction>
</xsd:simpleType>

```

```
<xsd:simpleType name="AddressTypeType">
  <xsd:annotation>
    <xsd:documentation xml:lang="pl">
      Typ definiujący typ (rodzaj) adresu. Możliwe wartości:
      [registered] - adres zameldowania
      [correspondence] - adres do korespondencji
      [billing] - adres na fakturze
      [residence] - adres zamieszkania lub pobytu
      [pit] - adres do rozliczeń podatkowych PIT
      [company] - adres siedziby firmy
      [other] - inny rodzaj adresu
    </xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="registered"/>
    <xsd:enumeration value="correspondence"/>
    <xsd:enumeration value="billing"/>
    <xsd:enumeration value="residence"/>
    <xsd:enumeration value="pit"/>
    <xsd:enumeration value="company"/>
    <xsd:enumeration value="other"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>
```