

Szczegółowy Opis Interfejsu Wymiany Danych

Spis treści:

| | | |
|-------|----------------------------------------------|----|
| 1 | Wstęp..... | 3 |
| 2 | Schemat wymiany danych: | 3 |
| 3 | Opis wymagań dla transmitowanych danych..... | 4 |
| 4 | Metody..... | 4 |
| 4.1 | Logowanie..... | 4 |
| 4.1.1 | Klucz sesyjny | 4 |
| 4.1.2 | Funkcja: Login()..... | 5 |
| 4.1.3 | Rozpoczęcie wymiany danych..... | 5 |
| 4.2 | GetVehicleList | 5 |
| 4.3 | GetVehicleById..... | 7 |
| 4.4 | GetVehicleListLastState | 9 |
| 4.5 | GetVehicleEvents | 11 |
| 4.6 | GetFuncModificationStatus | 14 |
| 4 | Opis typów danych..... | 16 |
| 5 | WSDL..... | 18 |

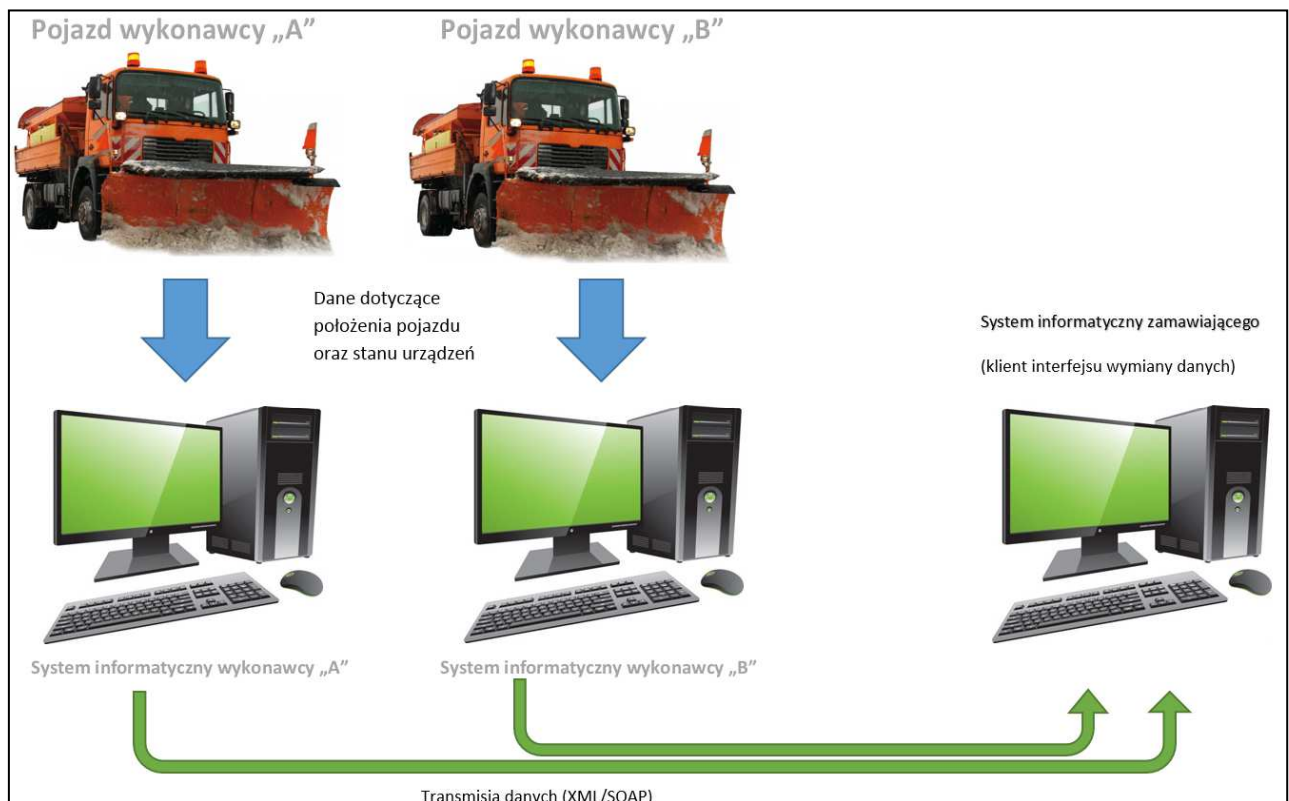
1 Wstęp

Niniejsza specyfikacja określa wymagania Zamawiającego dla zakresu danych otrzymywanych od operatorów dotyczących systemu utrzymania dróg oraz sposobu ich wymiany między systemami informatycznymi Wykonawcy i Zamawiającego.

Jako zakres wymaganych informacji transmitowanych poprzez interfejs wymiany danych, należy przyjąć zakres danych zawartych w poszczególnych metodach szczegółowo opisanych w kolejnych rozdziałach.

2 Schemat wymiany danych:

Celem wymiany danych jest przekazanie do Systemu Zamawiającego, danych zarejestrowanych przez pojazdy Wykonawców. Wszelkie rejestrowane dane i opisane w niniejszym dokumencie informacje, powinny być na bieżąco (w trybie online) przekazywane w oparciu o interfejs wymiany danych wykonany w technologii webservice. Transfer danych odbywa się za pomocą interfejsu wymiany danych opartym o usługę internetową udostępnioną przez Wykonawcę i działającym w oparciu o żądania HTTPS.



Rys. 1. Realizacja wymiany danych między Systemami wykonawców a zamawiającego

3 Opis wymagań dla transmitowanych danych

Transmisja danych z urządzeń o których mowa poniżej musi być jednoczesna z transmisją danych z systemu monitoringu GPS. Wszystkie zarejestrowane zdarzenia (Płuźenie, posypywanie, polewanie, zmiatanie) muszą być rozszerzone o dokładną datę i czas [zgodny z uniwersalnym czasem koordynowanym UTC(PL)] oraz współrzędne geograficzne zdarzeń wyznaczone na podstawie systemu GPS

Tabela 1. Wymagania odnośnie częstotliwości rejestracji danych przez pojazdy wykonawców.

| Typ zdarzenia | Moment Rejestracji |
|---------------------------------------------|------------------------------------------------------------------|
| Punkty Jazdy | Gdy pojazd jest w ruchu nie rzadziej niż co 100m lub co 5 sekund |
| Zmiana kierunku przemieszczania się pojazdu | Zmiana kąta o 30 stopni |
| Punkty postoju | Gdy pojazd stoi, nie rzadziej niż co 5 minut |
| Zdarzenie płuźenia | Natychmiast po wystąpieniu zdarzenia |
| Zdarzenie posypu | Natychmiast po wystąpieniu zdarzenia |
| Zdarzenia polewania | Natychmiast po wystąpieniu zdarzenia |
| Zdarzenie zmiatania | Natychmiast po wystąpieniu zdarzenia |

4 Metody

4.1 Logowanie

4.1.1 Klucz sesyjny

Celem zagwarantowania bezpieczeństwa na przesyłane przez wykonawców dane, wprowadzono autoryzację polegającą na wpisaniu nazwy użytkownika oraz hasła wraz z numerem Rejonu. Po uzupełnieniu danych pod warunkiem ich poprawności, zwracany jest tzw. klucz sesyjny, który jest niezbędny do wywołania poszczególnych funkcji (poza Login). Dzięki niemu uzyskiwany jest dostęp do danych Operatorów. W przypadku podania błędnego lub klucza który utracił ważność, wywoływana funkcja nie zwróci danych, lecz zasygnalizuje problem odpowiednim statusem.

Klucz sesyjny ma swój okres ważności. W przypadku, gdy przez pewien określony czas nie będzie wywołana żadna funkcja przy jego użyciu, zostanie uznany za nieważny.

4.1.2 Funkcja: Login()

Opis Funkcja zwraca klucz sesyjny wykorzystywany do komunikacji z *webservices* Operatora.

Wejście Nazwa użytkownika, hasło oraz identyfikator Rejonu:

User: string, Pass: string, CustomerNumber: Integer.

Wyjście Klucz sesyjny jako ciąg znaków (string) pozwalający na wywołanie pozostałych funkcji.

4.1.3 Rozpoczęcie wymiany danych

W celu ustalenia zakresu i ilości danych do pobrania z Systemu Operatora, System zamawiającego rozpoczyna wymianę danych od ustalenia momentu ostatniej zmiany danych w Systemie Operatora (w podziale na typy danych i dokumentów). Jeżeli data ostatniej zmiany danych jest późniejsza, niż data ostatniego pobrania danych, system zamawiającego pobiera nowe dane, wykorzystując stosowne funkcje.

4.2 GetVehicleList

Funkcja zwraca listę pojazdów używanych przez Operatora.

Wejście:

- Klucz sesyjny służący do uwierzytelnienia.
- ModifiedAfter: DateTime – data i czas od którego system powinien pobrać zmiany danych.

| Nazwa funkcji: | GetVehicleList |
|----------------|----------------------------------------|
| Opis: | Lista pojazdów. |
| Wyjście: | Lista pojazdów w postaci dokumentu XML |

```
<?xml version="1.0" encoding="utf-8" ?>
<result>
  <vehicle id="1">
    <name>Przykładowy pojazd</name>
    <plate-number>KR 12345</plate-number>
    <mark>FIAT</mark>
    <model>PANDA</model>
    <production-date>2006</production-date>
    <kerb-weight>1250</kerb-weight>
    <side-number>...</side-number>
    <last-modification>...</last-modification>
  </vehicle>
  <vehicle id="2">
    ...
  </vehicle>
  ...
  <vehicle id="N">
    ...
  </vehicle>
</result>
```

Rys. 2 Struktura pliku XML zwracanego przez funkcję GetVehicleList()

Tabela 2. Opis parametrów zwracanych przez wywołanie metody GetVehicleList.

| Parametr: | Opis parametru: |
|-------------------|-------------------------------------------|
| vehicle id | Identyfikator pojazdu |
| name | Nazwa |
| plate-number | Numer rejestracyjny |
| mark | Marka |
| model | Model |
| production-date | Data produkcji |
| kerb-weight | Masa własna |
| side-number | Numer boczny |
| last-modification | Data i czas ostatniej modyfikacji rekordu |

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="result">

    <xs:complexType>

      <xs:sequence>

        <xs:element name="vehicle">

          <xs:complexType>

            <xs:sequence>

              <xs:element type="xs:string" name="name"/>

              <xs:element type="xs:string" name="plate-number"/>

              <xs:element type="xs:string" name="mark"/>

              <xs:element type="xs:string" name="model"/>

              <xs:element type="xs:short" name="production-date"/>

              <xs:element type="xs:short" name="kerb-weight"/>

              <xs:element type="xs:string" name="side-number"/>

              <xs:element type="xs:datetime" name="last-modification"/>
            
```

```

        </xs:sequence>

        <xs:attribute type="xs:int" name="id" use="required"/>

    </xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

```

Rys. 3 Diagram XSD dla metody GetVehicleList

4.3 GetVehicleById

Funkcja zwraca parametry pojazdu o wskazanym identyfikatorze.

Wejście:

- Klucz sesyjny służący do uwierzytelnienia (SessionId)
- IDPojazdu: (VehicleID)

| Nazwa funkcji: | GetVehicleById |
|----------------|-----------------------------------------------|
| Opis: | Pobranie informacji tylko o jednym pojeździe. |
| Wyjście: | Opis pojazdu w postaci dokumentu XML |

```

<?xml version="1.0" encoding="utf-8" ?>
<result>
  <vehicle id="1">
    <name>Przykładowy pojazd</name>
    <plate-number>KR 12345</plate-number>
    <mark>FIAT</mark>
    <model>PANDA</model>
    <production-date>2006</production-date>
    <kerb-weight>1250</kerb-weight>
    <side-number>...</side-number>
    <last-modification>...</last-modification>
  </vehicle>
</result>

```

Rys. 4 Struktura pliku XML zwracanego przez funkcję GetVehicleById()

Tabela 3. Opis parametrów zwracanych przez wywołanie metody GetVehicleById

| Parametr: | Opis parametru: |
|-------------------|-------------------------------------------|
| vehicle id | Identyfikator pojazdu |
| name | Nazwa |
| plate-number | Numer rejestracyjny |
| mark | Marka |
| model | Model |
| production-date | Data produkcji |
| kerb-weight | Masa własna |
| side-number | Numer boczny |
| last-modification | Data i czas ostatniej modyfikacji rekordu |

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:element name="result">
```

```
    <xs:complexType>
```

```
      <xs:sequence>
```

```
        <xs:element name="vehicle">
```

```
          <xs:complexType>
```

```
            <xs:sequence>
```

```
              <xs:element type="xs:string" name="name"/>
```

```
              <xs:element type="xs:string" name="plate-number"/>
```

```
              <xs:element type="xs:string" name="mark"/>
```

```
              <xs:element type="xs:string" name="model"/>
```

```
              <xs:element type="xs:short" name="production-date"/>
```

```
              <xs:element type="xs:short" name="kerb-weight"/>
```

```
              <xs:element type="xs:string" name="side-number"/>
```

```
              <xs:element type="xs:datetime" name="last-modification"/>
```

```
            </xs:sequence>
```



```

        <xs:attribute type="xs:int" name="id" use="required"/>

        </xs:complexType>

    </xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

```

Rys 5. Diagram XSD dla metody GetVehicleById

4.4 GetVehicleListLastState

Funkcja zwraca informacje o bieżących pozycjach i stanach pojazdów.

| | |
|-----------------|-------------------------------------------------------------------------|
| Opis: | Informacje o bieżących pozycjach i stanu pojazdów – w formie XML |
| Wyjście: | XML opisujący poszczególny aktualny stan wszystkich pojazdów |

Wejście:

- Klucz sesyjny służący do uwierzytelnienia (SessionId)
- ModifiedAfter: DateTime data i czas od którego pobrać dane

```

<?xml version="1.0" encoding="utf-8" ?>
<result>
<vehicle-state id="1">
    <name>P1231</name>
    <datetime>2014-01-27 07:02:00</datetime>
    <country>PL</country>
    <gpscoordinates>
        <lat>52.3744234</lat>
        <long>16.2194345</long>
    </gpscoordinates>
    <state>jazda</state>
    <direction>282</direction>
    <velocity>9</velocity>
    <last-modification>2014-01-27 07:08:46</last-modification>
    <last-continous-data-event-modtime>2014-01-27 07:08:46</last-continous-data-event-modtime>
    <last-continous-data-event-time>2014-01-27 07:07:36</last-continous-data-event-time>
</vehicle-state>
</result>

```

Rys. 6 Struktura pliku XML zwracanego przez funkcję GetVehicleListLastState

Tabela 4. Opis parametrów zwracanych przez wywołanie metody GetVehicleListLastState

| Parametr: | Opis parametru: |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| name | Nazwa |
| Vehicle-state id | Identyfikator pojazdu |
| datetime | Data |
| country | Kraj |
| latitude | Latitude |
| longitude | Longitude |
| state | Stan- (jazda, postój)- ostatnie zarejestrowane zdarzenie |
| direction | Kierunek |
| velocity | Prędkość |
| last-modification | Data i czas ostatniej modyfikacji rekordu |
| last-continous-data-event-modtime | Wskazują datę i czas modyfikacji rekordu wskazanego przez pole Last-Continous-Data-Event-Time |
| last-continous-data-event-time | Wskazuje zarejestrowaną datę i czas wystąpienia ostatniego zdarzenia z ciągu zdarzeń które nie zostaną już zmodyfikowane. Oznacza to, że wszystkie zdarzenia zarejestrowane do tego czasu zostały już przesłane, nie występują już uzupełnienia i modyfikacje danych. Wszystkie dodane i zmodyfikowane w przyszłości rekordy będą miały datę i czas większy lub równy wskazanemu w tym polu |

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
<xs:element name="result">
```

```
<xs:complexType>
```

```
<xs:sequence>
```

```
<xs:element name="vehicle-state">
```

```
<xs:complexType>
```

```
<xs:sequence>
```

```
<xs:element type="xs:string" name="name" />
```

```
<xs:element type="xs:datetime" name="datetime" />
```

```

<xs:element type="xs:string" name="country"/>

<xs:element name="gpscoordinates">

  <xs:complexType>

    <xs:sequence>

      <xs:element type="xs:float" name="lat"/>

      <xs:element type="xs:float" name="long"/>

    </xs:sequence>

  </xs:complexType>

</xs:element>

<xs:element type="xs:string" name="state"/>

<xs:element type="xs:short" name="direction"/>

<xs:element type="xs:short" name="velocity"/>

<xs:element type="xs:string" name="last-modification"/>

<xs:element type="xs:datetime" name="last-continous-data-event-modtime"/>

<xs:element type="xs:datetime" name="last-continous-data-event-time"/>

</xs:sequence>

<xs:attribute type="xs:int" name="id"/>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

```

Rys. 7 Diagram XSD dla metody GetVehicleListLastState

4.5 GetVehicleEvents

Funkcja zwraca dane dotyczące pojazdu za zadany zakres czasu.

| | |
|-----------------|-------------------------------------------------------------|
| Opis: | Dane dla pojazdu za zadany zakres czasu. |
| Wyjście: | XML opisujący zdarzenia wykonane w zadanym przedziale czasu |

Wejście:

- Klucz sesyjny służący do uwierzytelnienia (SessionId)
- Identyfikator pojazdu (int, VehicleId)
- Data początkowa zakresu czasu dla danych(DateTime, DateTimeFrom)
- Data końcowa zakresu czasu dla danych (DateTime, DateTimeTo)
- ModifiedAfter: DateTime data i czas od którego pobrać dane

```
<?xml version="1.0" encoding="utf-8" ?>
<result update-date-time="2013-01-27 10:43:01">
  <event id="1">
    <vehicle-id>1</vehicle-id>
    <event-type-id>1</event-type-id>
    <gpscoordinates>
      <lat>52.0323</lat>
      <lon>19.3346</lon>
    </gpscoordinates>
    <startdatetime>2010-08-17 16:40:53</startdatetime>
    <enddatetime>2010-08-17 16:40:55</enddatetime>
    <maxspeed>...</maxspeed>
    <distance>0</distance>
    <direction>0</direction>
    <last-modification>2010-08-17 16:40:53</last-modification>
  </event>
</result>
```

Rys. 8 Struktura pliku XML zwracanego przez funkcję GetVehicleEvents

Tabela 5. Opis parametrów zwracanych przez wywołanie metody GetVehicleEvents

| Parametr: | Opis parametru: |
|-------------------|--------------------------------------------------------------------|
| vehicle-id | Identyfikator pojazdu |
| event id | unikalny identyfikator zdarzenia |
| event-type-id | Identyfikator typu zdarzenia (wg. Tabela5) |
| startdatetime | Data i czas rozpoczęcia zdarzenia |
| enddatetime | Data i czas zakończenia zdarzenia |
| gpscoordinates | (Latitude, Longitude) Długość i szerokość geograficzna |
| Maxspeed | Prędkość maksymalna od poprzedniego zarejestrowanego punktu w km/h |
| Distance | Dystans przebyty od ostatniego zdarzenia w metrach |
| Direction | Kierunek jazdy (azymut) w stopniach. |
| Last-modification | Data i czas ostatniej modyfikacji rekordu |

Tabela 6. Opis identyfikatorów zdarzeń zwracanych poprzez wywołanie metody GetVehicleEvents

| Event-Type-Id | Zdarzenie |
|---------------|-------------|
| 1 | Jazda |
| 2 | postój |
| 3 | płużenie |
| 4 | posypywanie |
| 5 | polewanie |
| 6 | zamiatanie |

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:element name="result">
```

```
    <xs:complexType>
```

```
      <xs:sequence>
```

```
        <xs:element name="event">
```

```
          <xs:complexType>
```

```
            <xs:sequence>
```

```
              <xs:element type="xs:int" name="vehicle-id"/>
```

```
              <xs:element type="xs:long" name="event-type-id"/>
```

```
              <xs:element name="gpscoordinates">
```

```
                <xs:complexType>
```

```
                  <xs:sequence>
```

```
                    <xs:element type="xs:float" name="lat"/>
```

```
                    <xs:element type="xs:float" name="long"/>
```

```
                  </xs:sequence>
```

```

        </xs:complexType>

        </xs:element>

        <xs:element type="xs:datetime" name="startdate" />

        <xs:element type="xs:datetime" name="enddate" />

        <xs:element type="xs:byte" name="maxspeed" />

        <xs:element type="xs:short" name="distance" />

        <xs:element type="xs:byte" name="direction" />

        <xs:element type="xs:datetime" name="last-modification" />

    </xs:sequence>

    <xs:attribute type="xs:byte" name="id" />

</xs:complexType>

</xs:element>

</xs:sequence>

<xs:attribute type="xs:string" name="update-date-time" />

</xs:complexType>

</xs:element>

</xs:schema>

```

Rys. 9 Diagram XSD dla metody GetVehicleEvents

4.6 GetFuncModificationStatus

| | |
|-----------------------|-----------------------------------------------------------------------------------------------|
| Nazwa funkcji: | GetFuncModificationStatus |
| Opis: | Pobranie informacji o ostatniej aktualizacji danych dostępnych przez interfejsy do integracji |
| Wyjście: | Lista funkcji ze statusem modyfikacji w postaci dokumentu XML |

Wejście:

- Klucz sesyjny służący do uwierzytelnienia (SessionId)
- ModifiedAfter: DateTime data i czas od którego pobrać dane

```
<?xml version="1.0" encoding="utf-8" ?>
<result>
  <functions>
    <function>
      <name>GetVehicleList</name>
      <last-modification>2014-12-14 15:46:20</last-modification>
    </function>
    <function>
      <name>GetVehicleEvents</name>
      <last-modification>2014-11-14 15:46:20</last-modification>
    </function>
    ...
    ...
  </functions>
</result>
```

Rys. 10 Struktura pliku XML zwracanego przez funkcję GetFuncModificationStatus

Tabela 7. Opis parametrów zwracanych przez wywołanie metody GetFuncModificationStatus

| Parametr: | Opis parametru: |
|-------------------|------------------|
| Name | Nazwa metody |
| Last-modification | Data modyfikacji |

```
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:element name="result">

    <xs:complexType>

      <xs:sequence>

        <xs:element name="functions">

          <xs:complexType>

            <xs:sequence>

              <xs:element name="function">

                <xs:complexType>

                  <xs:sequence>

                    <xs:element type="xs:string" name="name" />
```

```

        <xs:element type="xs:string" name="last-modification"/>

    </xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:sequence>

</xs:complexType>

</xs:element>

</xs:schema>

```

Rys. 11 Diagram XSD dla metody GetFuncModificationStatus

4. Opis typów danych

Tabela 8. Opis typów danych.

| Parametr: | Opis parametru: | Typ |
|-------------------|-------------------------------------------|----------|
| vehicle id | Identyfikator pojazdu | int |
| name | Nazwa pojazdu | string |
| plate-number | Numer rejestracyjny | string |
| mark | Marka | string |
| model | Model | string |
| production-date | Data produkcji | short |
| kerb-weight | Masa własna | short |
| side-number | Numer boczny | string |
| last-modification | Data i czas ostatniej modyfikacji rekordu | datetime |
| datetime | Data | datetime |
| country | Kraj | string |

| | | |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------|
| latitude | Latitude | double |
| longitude | Longitude | double |
| state | Stan- (jazda, postój)- ostatnie zarejestrowane zdarzenie | string |
| direction | Kierunek | double |
| velocity | Prędkość | double |
| last-modification | Data i czas ostatniej modyfikacji rekordu | datetime |
| last-continous-data-event-modtime | Wskazują datę i czas modyfikacji rekordu wskazanego przez pole Last-Continous-Data-Event-Time | datetime |
| last-continous-data-event-time | Wskazuje zarejestrowaną datę i czas wystąpienia ostatniego zdarzenia z ciągu zdarzeń które nie zostaną już zmodyfikowane. Oznacza to, że wszystkie zdarzenia zarejestrowane do tego czasu zostały już przesłane, nie występują już uzupełnienia i modyfikacje danych. Wszystkie dodane i zmodyfikowane w przyszłości rekordy będą miały datę i czas większy lub równy wskazanemu w tym polu | datetime |
| vehicle-id | Identyfikator pojazdu | int |
| event id | unikalny identyfikator zdarzenia | long |
| event-type-id | Identyfikator typu zdarzenia (wg. Tabela5) | int |
| startdatetime | Data i czas rozpoczęcia zdarzenia | datetime |
| enddatetime | Data i czas zakończenia zdarzenia | datetime |
| Maxspeed | Prędkość maksymalna od poprzedniego zarejestrowanego punktu w km/h | short |
| Distance | Dystans przebyty od ostatniego zdarzenia w metrach | short |
| Direction | Kierunek jazdy (azymut) w stopniach. | short |
| Last-modification | Data i czas ostatniej modyfikacji rekordu | datetime |

5. WSDL

```
<wSDL:definitions xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://localhost/" xmlns:intf="http://localhost/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:tns1="urn:uXMLRemotable"
xmlns:wSDL="http://schemas.xmlsoap.org/wSDL/"
xmlns:wSDLsoap="http://schemas.xmlsoap.org/wSDL/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" targetNamespace="http://localhost/">
<wSDL:types>
<schema xmlns="http://www.w3.org/2001/XMLSchema" targetNamespace="urn:uXMLRemotable">
<import namespace="http://schemas.xmlsoap.org/soap/encoding/" />
<complexType name="TXMLRemotable">
<sequence>
<element name="Status" type="xsd:int" />
<element name="XML" type="xsd:string" />
</sequence>
</complexType>
</schema>
</wSDL:types>
<wSDL:message name="GetVehicleListLastStateRequest">
<wSDL:part name="SessionId" type="xsd:string"></wSDL:part>
<wSDL:part name="ModifiedAfter" type="xsd:dateTime"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleListResponse">
<wSDL:part name="return" type="tns1:TXMLRemotable"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleByIdRequest">
<wSDL:part name="SessionId" type="xsd:string"></wSDL:part>
<wSDL:part name="VehicleID" type="xsd:int"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleListRequest">
<wSDL:part name="SessionId" type="xsd:string"></wSDL:part>
<wSDL:part name="ModifiedAfter" type="xsd:dateTime"></wSDL:part>
</wSDL:message>
<wSDL:message name="LoginResponse">
<wSDL:part name="return" type="xsd:string"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleEventsResponse">
<wSDL:part name="return" type="tns1:TXMLRemotable"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleEventsRequest">
<wSDL:part name="SessionId" type="xsd:string"></wSDL:part>
<wSDL:part name="VehicleId" type="xsd:int"></wSDL:part>
<wSDL:part name="ModifiedAfter" type="xsd:dateTime"></wSDL:part>
<wSDL:part name="DateFrom" type="xsd:dateTime"></wSDL:part>
<wSDL:part name="DateTo" type="xsd:dateTime"></wSDL:part>
</wSDL:message>
<wSDL:message name="LoginRequest">
<wSDL:part name="User" type="xsd:string"></wSDL:part>
<wSDL:part name="Password" type="xsd:string"></wSDL:part>
<wSDL:part name="CustomerNumber" type="xsd:int"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleByIdResponse">
<wSDL:part name="return" type="tns1:TXMLRemotable"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetVehicleListLastStateResponse">
<wSDL:part name="return" type="tns1:TXMLRemotable"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetFuncModificationStatusResponse">
<wSDL:part name="return" type="tns1:TXMLRemotable"></wSDL:part>
</wSDL:message>
<wSDL:message name="GetFuncModificationStatusRequest">
```

```

<wsdl:part name="SessionId" type="xsd:string"></wsdl:part>
</wsdl:message>
<wsdl:portType name="IAZWebService">
  <wsdl:operation name="Login" parameterOrder="User Password CustomerNumber">
    <wsdl:input message="impl:LoginRequest" name="LoginRequest"></wsdl:input>
    <wsdl:output message="impl:LoginResponse" name="LoginResponse"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetVehicleList" parameterOrder="SessionId ModifiedAfter">
    <wsdl:input message="impl:GetVehicleListRequest" name="GetVehicleListRequest"></wsdl:input>
    <wsdl:output message="impl:GetVehicleListResponse"
name="GetVehicleListResponse"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetVehicleById" parameterOrder="SessionId VehicleID">
    <wsdl:input message="impl:GetVehicleByIdRequest" name="GetVehicleByIdRequest"></wsdl:input>
    <wsdl:output message="impl:GetVehicleByIdResponse"
name="GetVehicleByIdResponse"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetVehicleListLastState" parameterOrder="SessionId ModifiedAfter">
    <wsdl:input message="impl:GetVehicleListLastStateRequest"
name="GetVehicleListLastStateRequest"></wsdl:input>
    <wsdl:output message="impl:GetVehicleListLastStateResponse"
name="GetVehicleListLastStateResponse"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetVehicleEvents" parameterOrder="SessionId VehicleId ModifiedAfter
DateFrom DateTo">
    <wsdl:input message="impl:GetVehicleEventsRequest"
name="GetVehicleEventsRequest"></wsdl:input>
    <wsdl:output message="impl:GetVehicleEventsResponse"
name="GetVehicleEventsResponse"></wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetFuncModificationStatus" parameterOrder="SessionId">
    <wsdl:input message="impl:GetFuncModificationStatusRequest"
name="GetFuncModificationStatusRequest"></wsdl:input>
    <wsdl:output message="impl:GetFuncModificationStatusResponse"
name="GetFuncModificationStatusResponse"></wsdl:output>
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="IAZWebServicePortSoapBinding" type="impl:IAZWebService">
  <wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="Login">
    <wsdlsoap:operation soapAction="urn:AZWebServiceIntf-IAZWebService#Login"/>
    <wsdl:input name="LoginRequest">
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:AZWebServiceIntf-IAZWebService" use="encoded"/>
    </wsdl:input>
    <wsdl:output name="LoginResponse">
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://localhost/" use="encoded"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetVehicleList">
    <wsdlsoap:operation soapAction="urn:AZWebServiceIntf-IAZWebService#GetVehicleList"/>
    <wsdl:input name="GetVehicleListRequest">
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:AZWebServiceIntf-IAZWebService" use="encoded"/>
    </wsdl:input>
    <wsdl:output name="GetVehicleListResponse">
      <wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://localhost/" use="encoded"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetVehicleById">

```

```

<wsdl:soap:operation soapAction="urn:AZWebServiceIntf-IAZWebService#GetVehicleById"/>
<wsdl:input name="GetVehicleByIdRequest">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:AZWebServiceIntf-IAZWebService" use="encoded"/>
</wsdl:input>
<wsdl:output name="GetVehicleByIdResponse">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://localhost/" use="encoded"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetVehicleListLastState">
<wsdl:soap:operation soapAction="urn:AZWebServiceIntf-
IAZWebService#GetVehicleListLastState"/>
<wsdl:input name="GetVehicleListLastStateRequest">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:AZWebServiceIntf-IAZWebService" use="encoded"/>
</wsdl:input>
<wsdl:output name="GetVehicleListLastStateResponse">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://localhost/" use="encoded"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetVehicleEvents">
<wsdl:soap:operation soapAction="urn:AZWebServiceIntf-IAZWebService#GetVehicleEvents"/>
<wsdl:input name="GetVehicleEventsRequest">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:AZWebServiceIntf-IAZWebService" use="encoded"/>
</wsdl:input>
<wsdl:output name="GetVehicleEventsResponse">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://localhost/" use="encoded"/>
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetFuncModificationStatus">
<wsdl:soap:operation soapAction="urn:AZWebServiceIntf-
IAZWebService#GetFuncModificationStatus"/>
<wsdl:input name="GetFuncModificationStatusRequest">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="urn:AZWebServiceIntf-IAZWebService" use="encoded"/>
</wsdl:input>
<wsdl:output name="GetFuncModificationStatusResponse">
<wsdl:soap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
namespace="http://localhost/" use="encoded"/>
</wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="IAZWebServiceservice">
<wsdl:port binding="impl:IAZWebServicePortSoapBinding" name="IAZWebServicePort">
<wsdl:soap:address location="https://AZ.pl/GPSExportAZ_Server/services/IAZWebServicePort"/>
</wsdl:port>
</wsdl:service>
</wsdl:definitions>

```