



New Logic Booking API

Developer's guide

v.2.8

Language – English

Copyright © 2012 - 2019 – New Logic

Annotation

This API makes it possible to carry out the following operations:

- getting static data (list of countries, regions, cities, hotels, excursions, transfer facilities, hotel rooms, currencies, room types, types of meals and placement, order statuses)
- search of hotels
- getting prices for early check-in and late check-out
- search of excursions
- search of transfer facilities
- ordering
- search of orders
- getting information about the order
- order modification
- order confirmation
- order cancellation

New Logic Booking API

| Version | Date | Description | Pages |
|---------|------------|--|-------|
| 1.0 | 24.01.2012 | First release of the guide, product's update. | 36 |
| 1.1 | 07.02.2013 | Added display of price of penalties for change and cancellation order when getting order information. | 37 |
| 1.2 | 25.02.2013 | Added price detailing for each day by room when search of hotels. | 38 |
| 1.3 | 02.04.2013 | Added display of rules of booking and cancellation, date of penalties, penalty amount in case of cancellation when search of content for the order. Added cancellation policy when getting information about the order. | 38 |
| 1.4 | 04.04.2013 | Added additional options when getting information about the types of placement. Added display of room types when search of hotels. | 38 |
| 1.5 | 20.06.2013 | Added display of hotel address and hotel coordinates when getting the list of hotels. | 38 |
| 1.6 | 22.08.2013 | Integrated special action tariffs. Changed the structure of search of hotels, adding and editing orders, getting order information. | 41 |
| 1.7 | 11.09.2013 | Added the information about the order status to the GetOrderInfo method. | 42 |
| 1.8 | 11.10.2013 | Added the possibility to search by hotel. | 42 |
| 1.9 | 17.10.2013 | Added the possibility to search rooms by availability status to the SearchHotel method. | 42 |
| 2.0 | 13.01.2014 | Added the possibility to search rooms without detailing prices for days to the method SearchHotel. It is possible to get only a recommended variant from the hotel. | 42 |
| 2.1 | 18.02.2014 | The following parameters were added to the response of method HotelList: class of the hotel, time of arrival and time of departure. | 42 |
| 2.2 | 04.02.2014 | Added description of the method to get the list of hotel classes HotelClassList. | 43 |
| 2.3 | 05.08.2014 | Added the possibility to get the number of available allotments in method SearchHotel. The request is the same, but the response is containing one more parameter. | 45 |
| 2.4 | 13.05.2015 | Added method for search orders by date of booking and the hotel SearchOrders. Updated guide: - formatted text - added information about the available languages - in response examples of method SearchHotelRS removed attributes «id», «href» in elements «Access», «BookingPolicyInfo», «CancellationPolicyInfo» (typo in the documentation) - in response example of method SearchExcursionRS changed date format in element «DateExcursion» (typo | 35 |

New Logic Booking API

| | | | |
|-----|------------|--|----|
| | | <p>in the documentation)</p> <ul style="list-style-type: none"> - in description of elements method RoomListRS changed name of element «meals type» – «FoodType» (was «FoodTypeS» – typo in the documentation) - in request example of method AddOrderRQ changed name of method (was «AddOrderRequest» – typo in the documentation) - in response example of method AddOrderRS changed name of method (was «AddOrderResponse» – typo in the documentation) - added description of new method SearchOrders - in description of elements method EditOrderRQ removed attributes «id», «name» for element «RoomActionTariff» (typo in the documentation) - in response example of method EditOrderRS changed name of method (was «AddOrderResponse» – typo in the documentation) | |
| 2.5 | 08.07.2015 | Added method for order confirmation ConfirmOrder. | 36 |
| 2.6 | 25.08.2015 | Added method for getting prices for early check-in and late check-out GetOvertime. In methods AddOrderRQ and EditOrderRQ in element «Option» added attribute «TypeEarlyCheckIn». | 39 |
| 2.7 | 14.11.2018 | API renamed to «New Logic Booking API». Updated guide: - in request example of method TransferListRQ changed name of method (was «AddServiceListRQ» – typo in the documentation) | 40 |
| 2.8 | 04.07.2019 | Added method for getting hotel surcharges GetHotelSurcharges. In response of method GetOrderInfoRS added element «HotelSurcharge». Added method for simulating order modification SimulateEditOrder. | 42 |

Table of contents

| | | |
|------|---|----|
| 1 | General information | 6 |
| 2 | Getting static data | 7 |
| 2.1 | Getting the list of countries..... | 7 |
| 2.2 | Getting the list of regions | 7 |
| 2.3 | Getting the list of cities..... | 8 |
| 2.4 | Getting the list of hotels | 9 |
| 2.5 | Getting the list of excursions..... | 10 |
| 2.6 | Getting the list of transfers | 11 |
| 2.7 | Getting the list of hotel rooms | 12 |
| 2.8 | Getting the list of hotel room types | 13 |
| 2.9 | Getting the list of placement types | 13 |
| 2.10 | Getting the list of meal types | 14 |
| 2.11 | Getting the list of currencies | 15 |
| 2.12 | Getting the list of hotel classes | 15 |
| 2.13 | Getting the list of order statuses..... | 16 |
| 3 | Search of content for order | 18 |
| 3.1 | Search of hotels | 18 |
| 3.2 | Getting prices for early check-in and late check-out..... | 22 |
| 3.3 | Getting the list of hotel surcharges | 24 |
| 3.4 | Search of excursions | 26 |
| 3.5 | Search of transfers..... | 27 |
| 4 | Operations with orders..... | 29 |
| 4.1 | Ordering..... | 29 |
| 4.2 | Search of orders..... | 31 |
| 4.3 | Getting information about the order | 32 |
| 4.4 | Order modification | 35 |
| 4.5 | Simulating order modification..... | 37 |
| 4.6 | Order confirmation..... | 39 |
| 4.7 | Order cancellation | 40 |
| 5 | Contacts..... | 42 |

1 General information

The exchange of data between a client and a server is made with the help of the SOAP protocol. Every user receives a user name, password and a range of IP addresses for connection with the server. The obligatory element of each request is `<Auth login="test@test.com" pass="123"/>`, which contains the user name and password.

In order to decrease the size of the transferred XML files, the replacement of the static data with the corresponding ID is used. For instance, for receiving a list of all the hotels in a specified city, it is necessary to transfer not the name of the city but it's ID. The process of getting all the static data is described in details in the corresponding parts of this document.

http://www.newlogic.ua/module/api/get_wsdl/ – address of wsdl scheme for connection

http://test2.newlogic.ua/module/api/get_wsdl/ – address of wsdl scheme for testing

For getting text information in the desired language is necessary in each request to specify the attribute lang. Available languages:

- ua – Ukrainian
- en – English
- ru – Russian

Example for initialization of service and getting the list of countries (programming language is PHP):

```
$client = new SoapClient('http://www.newlogic.ua/module/api/get_wsdl/');
$params = array(
    'lang' => 'en',
    'Auth' => array(
        'login' => 'test@test.com',
        'pass' => '123'
    )
);
$RS = $client->GetCountryList($params);
```

2 Getting static data

This part explains how to get necessary static information about countries, regions, cities, hotels, excursions, transfers, types of meals and placement.

2.1 Getting the list of countries

Description:

Getting all active countries of the system. The output is a list of countries that have the attributes:

- id – unique country identifier
- name – country name on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CountryListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </CountryListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CountryListRS>
      <Countries>
        <Country id="30" name="Ukraine"/>
      </Countries>
    </CountryListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

2.2 Getting the list of regions

Description:

Getting regions from the selected country. The region has attributes:

- id – unique region identifier
- name – region name on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <RegionListRQ lang="en" country="30">
      <Auth login="test@test.com" pass="123"/>
    </RegionListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <RegionListRS>
      <Regions>
        <Region id="47" name="Crimea"/>
        <Region id="46" name="Dnipropetrovsk"/>
        <Region id="44" name="Donetsk"/>
      </Regions>
    </RegionListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Provided above example demonstrates selection of all regions of the country with «id» – 30. The language of representation is «en».

2.3 Getting the list of cities

Description:

Getting cities from the region or the country. The city has attributes:

- id – unique city identifier
- name – city name on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CityListRQ lang="en" country="30" region="38">
      <Auth login="test@test.com" pass="123"/>
    </CityListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CityListRS>
      <Cities>
        <City id="8912" name="Bila Tserkva"/>
      </Cities>
    </CityListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```



```
<City id="29" name="Kiev"/>
</Cities>
</CityListRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Provided above example demonstrates selection of all cities from the country with «id» – 30 and the region with «id» – 38. The language of selection is «en».

2.4 Getting the list of hotels

Description:

Getting hotels from selected city. Hotel has attributes:

- id – unique hotel identifier
- name – hotel name on the specified language
- class – identifier of hotel class (stars), which is the mapping with the method «getHotelClassList»

Elements:

- TimeArrival – check-in date
- TimeDeparture – check-out date
- Description – hotel description on the specified language
- Address – hotel address on the specified language
- Latitude – hotel’s geographic coordinates (latitude)
- Longitude – hotel’s geographic coordinates (longitude)
- Photo – hotel’s photo
- Hotelservice – list of services, provided in the hotel

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <HotelListRQ lang="en" city="40">
      <Auth login="test@test.com" pass="123"/>
    </HotelListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <HotelListRS>
      <Hotels>
        <HotelItem id="1" name="Hotel name" class="11">
          <TimeArrival>14:00</TimeArrival>
        </HotelItem>
      </Hotels>
    </HotelListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```
<TimeDeparture>12:00</TimeDeparture>
<Description>Some description.</Description>
<Address>9, Minskaya Str.</Address>
<Latitude>47.865961451122</Latitude>
<Longitude>35.054122209549</Longitude>
<Photo>http://test/245f4e11.jpg</Photo>
<Hotelservice>24 Hour Room Service</Hotelservice>
</HotelItem>
</Hotels>
</HotelListRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

This example demonstrates the selection of hotels in the city with «id» – 40. The language is «en».

2.5 Getting the list of excursions

Description:

Getting excursions from selected city. Excursion has attributes:

- id – unique excursion identifier
- name – excursion name on the specified language

Elements:

- Description – excursion description on the specified language
- Photo – excursion's photo

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <AddServiceListRQ lang="en" city="29">
      <Auth login="test@test.com" pass="123"/>
    </AddServiceListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <AddServiceListRS>
      <AddServices>
        <AddService id="66" name="Excursion">
          <Description>Description.</Description>
          <Photo>http://test/0dbe7712.jpg</Photo>
        </AddService>
      </AddServices>
    </AddServiceListRS>
```

```
</SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```

This example demonstrates the selection of excursions in the city with «id» – 29. The language is «en».

2.6 Getting the list of transfers

Description:

Getting transfers from selected city. Transfer has attributes:

- id – unique transfer identifier
- name – transfer name on the specified language

Elements:

- Description – transfer description on the specified language
- Photo – transfer's photo

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>  
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">  
  <SOAP-ENV:Body>  
    <TransferListRQ lang="en" city="29">  
      <Auth login="test@test.com" pass="123"/>  
    </TransferListRQ>  
  </SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>  
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">  
  <SOAP-ENV:Body>  
    <TransferListRS>  
      <Transfers>  
        <Transfer id="38" name="Transfer">  
          <Description>Transfer description.</Description>  
          <Photo>http://test/6b8c7d5d.jpg</Photo>  
        </Transfer>  
      </Transfers>  
    </TransferListRS>  
  </SOAP-ENV:Body>  
</SOAP-ENV:Envelope>
```

This example demonstrates the selection of transfers in the city with «id» – 29. The language is «en».

2.7 Getting the list of hotel rooms

Description:

Getting rooms from the selected hotel. Room has attributes:

- id – unique room identifier
- name – room name on the specified language

Elements:

- RoomType – identifier and name of room type
- Description – room description on the specified language
- Photo – room's photo
- FoodType – type of meals
- Placement – type of placement

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <RoomListRQ lang="en" hotel="80">
      <Auth login="test@test.com" pass="123"/>
    </RoomListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <RoomListRS>
      <Rooms>
        <Room id="36" name="Double Economy">
          <RoomType id="9" name="DBL/TWN"/>
          <Description>description</Description>
          <Photo>http://test/9da47f64.jpg</Photo>
          <FoodType id="15" name="BB"/>
          <Placement id="7" name="1"/>
          <Placement id="8" name="2"/>
        </Room>
      </Rooms>
    </RoomListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

This example demonstrates the selection of hotel rooms in the hotel with «id» – 80. The language is «en».

2.8 Getting the list of hotel room types

Description:

Getting room types. Room type has attributes:

- id – unique room type identifier
- name – room type name on the specified language

Request example:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <RoomTypeListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </RoomTypeListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <RoomTypeListRS>
      <RoomTypes>
        <RoomType id="8" name="DBL"/>
        <RoomType id="9" name="DBL/TWN"/>
      </RoomTypes>
    </RoomTypeListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

2.9 Getting the list of placement types

Description:

Getting types of placement. Type of placement has attributes:

- id – unique placement type identifier
- name – placement type name on the specified language
- adult – number of adults for this type of placement
- children – number of children for this type of placement

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <PlacementListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </PlacementListRQ>
  </SOAP-ENV:Body>
```

</SOAP-ENV:Envelope>

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <PlacementListRS>
      <Placements>
        <Placement id="7" name="1" adult="1" children="0"/>
        <Placement id="9" name="1+1" adult="1" children="1"/>
      </Placements>
    </PlacementListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

2.10 Getting the list of meal types

Description:

Getting types of meal. Type of meal has attributes:

- id – unique meal type identifier
- name – meal type name on the specified language

Elements:

- Description – meal type description on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <FoodTypeListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </FoodTypeListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <FoodTypeListRS>
      <FoodTypes>
        <FoodType id="15" name="BB">
          <Description>Bed and breakfast</Description>
        </FoodType>
      </FoodTypeListRS>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

2.11 Getting the list of currencies

Description:

Getting currencies. Currency has attributes:

- id – unique currency identifier
- name – currency name

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CurrencyListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </CurrencyListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CurrencyListRS>
      <Currencies>
        <Currency id="17" name="UAH"/>
        <Currency id="18" name="USD"/>
      </Currencies>
    </CurrencyListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

2.12 Getting the list of hotel classes

Description:

Getting hotel classes. Hotel class has attributes:

- id – unique hotel class identifier
- name – hotel class name

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <HotelClassListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </HotelClassListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <HotelClassListRS>
      <HotelClasses>
        <HotelClass id="15" name="1*"/>
        <HotelClass id="13" name="2*"/>
        <HotelClass id="10" name="3*"/>
        <HotelClass id="11" name="4*"/>
        <HotelClass id="12" name="5*"/>
      </HotelClasses>
    </HotelClassListRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

2.13 Getting the list of order statuses

Description:

Getting statuses of order. Order status has attributes:

- id – unique order status identifier
- name – order status name

Elements:

- Description – order status description on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <StatusListRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
    </StatusListRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <StatusListRS>
      <Statuses>
        <Status id="3" name="modified">
          <Description>Order modified.Need reconfirmation by provider.</Description>
        </Status>
      </Statuses>
    </StatusListRS>
  </SOAP-ENV:Body>
```


</SOAP-ENV:Envelope>

3 Search of content for order

3.1 Search of hotels

Description:

Search of hotels on the set parameters. Request attributes:

- lang – language for getting data (the parameter is optional)

Request elements (search parameters):

- City – city identifier
- Hotel – hotel identifier
- Available – availability of the hotel rooms. Valid values:
 - online – available only for instant booking
 - onrequest – available only on request
 - all – all hotel rooms
- PriceDetail – detailed price for each day. This option is disabled by default. Valid values:
 - 0 – disabled detailed price for each day. If detailed price for each day is off, in the response comes only recommended variant from the hotel
 - 1 – enabled detailed price for each day. If detailed price for each day is on, in the response comes all the rooms of the hotel with the parameter Available, each room has a breakdown of prices by days
- Currency – currency identifier
- Date – period of stay. Attributes:
 - from – check-in date
 - to – check-out date
- Adult – number of adults
- Children – number of children with the specification of children age

As a result of the search arrives the data packet at the following structure. Response attributes and elements:

- Hotel – container hotel. Includes a plurality of containers Room. Attributes:
 - id – hotel identifier
- Room – container hotel room. It contains information about the hotel room. Attributes:
 - id – hotel room identifier
- RoomActionTariff – action tariff. Attributes:
 - id – unique action tariff identifier
 - name – action tariff name
- RoomType – hotel room type. Attributes:
 - id – unique hotel room type identifier
 - name – hotel room type name
- Placement – type of placement in hotel room. Attributes:
 - id – placement type identifier
- FoodType – type of meal in hotel room. Attributes:
 - id – meal type identifier
- Price – price for hotel room. Currency of price is currency that has been specified in the

- request parameters
- PricePerDay – prices for room per each day. This element is included in a response only if in a request was set price detailing per each day (parameter PriceDetail = 1). It contains many containers PricePerDayDetails
 - PricePerDayDetails – price detailing per room for each day. Attributes:
 - Date – date
 - Price – price
 - Access – room availability on current day
 - Allotment – count of available allotments on current day
 - Access – room availability. It indicates that the room is available for instant booking or available on request
 - Allotment – count of available allotments
 - CancellationPolicy – policy penalties on cancellation. Elements:
 - Price – penalty price
 - DateTimeFrom – date of penalty begin
 - BookingPolicyInfo – information about reservation rules on the specified language
 - CancellationPolicyInfo – information about cancellation rules on the specified language

Request example for search by city:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchHotelRQ lang="ru">
      <Auth login="test@test.com" pass="123"/>
      <City id="29"/>
      <Hotel id="0"/>
      <Available>all</Available>
      <Currency id="17"/>
      <Date from="2014-08-13" to="2014-08-16"/>
      <Adult count="1"/>
      <Children/>
      <PriceDetail>0</PriceDetail>
    </SearchHotelRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example for search by city:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchHotelRS>
      <Hotel id="185">
        <Room id="313">
          <RoomActionTariff id="0" name=""/>
          <RoomType id="9" name="DBL/TWN"/>
          <Placement id="7"/>
          <FoodType id="15"/>
          <Price>3030</Price>
          <Access>Номера доступны для моментального бронирования</Access>
          <Allotment>10</Allotment>
        </Room>
      </Hotel>
    </SearchHotelRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

<CancellationPolicy>
  <Price>0</Price>
  <DateTimeFrom>2014-08-11 12:00:00</DateTimeFrom>
</CancellationPolicy>
  <BookingPolicyInfo>Бронирование является гарантированным до 18:00 в день заезда. Пребывающие
  после 18:00 гости должны уведомить об этом в поле «Примечания» во время бронирования или сообщением по
  электронной почте, с указанием номера бронирования до 14:00 дня заезда. В противном случае, New Logic имеет
  право отменить бронирование на условиях незаезда (см. Условия аннуляции).</BookingPolicyInfo>
  <CancellationPolicyInfo>Заказ может быть аннулирован бесплатно до времени наступления штрафных
  санкций. В случае поздней аннуляции или незаезда гостя система бронирования взимает 100% стоимости номера
  за 1 сутки.</CancellationPolicyInfo>
</Room>
</Hotel>
<Hotel id="160">
  <Room id="375">
    <RoomActionTariff id="0" name=""/>
    <RoomType id="8" name="DBL"/>
    <Placement id="7"/>
    <FoodType id="15"/>
    <Price>3501</Price>
    <Access>Номера доступны по запросу</Access>
    <Allotment>10</Allotment>
    <CancellationPolicy>
      <Price>0</Price>
      <DateTimeFrom>2014-08-12 14:00:00</DateTimeFrom>
    </CancellationPolicy>
    <BookingPolicyInfo>Бронирование является гарантированным до 18:00 в день заезда. Пребывающие
    после 18:00 гости должны уведомить об этом в поле «Примечания» во время бронирования или сообщением по
    электронной почте, с указанием номера бронирования до 14:00 дня заезда. В противном случае, New Logic имеет
    право отменить бронирование на условиях незаезда (см. Условия аннуляции).</BookingPolicyInfo>
    <CancellationPolicyInfo>Заказ может быть аннулирован бесплатно до времени наступления штрафных
    санкций. В случае поздней аннуляции или незаезда гостя система бронирования взимает 100% стоимости номера
    за 1 сутки.</CancellationPolicyInfo>
  </Room>
</Hotel>
</SearchHotelRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Request example for search by hotel:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchHotelRQ lang="ru">
      <Auth login="test@test.com" pass="123"/>
      <City id="0"/>
      <Hotel id="80"/>
      <Currency id="19"/>
      <Date from="2013-10-27" to="2013-10-28"/>
      <Adult count="1"/>
      <Children>
        <Child age="4"/>
        <Child age="6"/>
      </Children>
      <PriceDetail>1</PriceDetail>
    </SearchHotelRQ>
  </SOAP-ENV:Body>

```

 </SOAP-ENV:Envelope>
Response example for search by hotel:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchHotelRS>
      <Hotel id="300">
        <Room id="1218">
          <RoomActionTariff id="0" name=""/>
          <RoomType id="7" name="SGL"/>
          <Placement id="7"/>
          <FoodType id="19"/>
          <Price>4002</Price>
          <PricePerDay>
            <PricePerDayDetails>
              <Date>2014-08-13</Date>
              <Price>1334</Price>
              <Access>Номера доступны по запросу</Access>
              <Allotment>10</Allotment>
            </PricePerDayDetails>
            <PricePerDayDetails>
              <Date>2014-08-14</Date>
              <Price>1334</Price>
              <Access>Номера доступны по запросу</Access>
              <Allotment>10</Allotment>
            </PricePerDayDetails>
            <PricePerDayDetails>
              <Date>2014-08-15</Date>
              <Price>1334</Price>
              <Access>Номера доступны по запросу</Access>
              <Allotment>10</Allotment>
            </PricePerDayDetails>
          </PricePerDay>
          <Access>Номера доступны по запросу</Access>
          <Allotment>10</Allotment>
          <CancellationPolicy>
            <Price>1334</Price>
            <DateTimeFrom>2014-08-10 14:00:00</DateTimeFrom>
          </CancellationPolicy>
          <BookingPolicyInfo>Бронирование является гарантированным до 18:00 в день заезда. Пребывающие после 18:00 гости должны уведомить об этом в поле «Примечания» во время бронирования или сообщением по электронной почте, с указанием номера бронирования до 14:00 дня заезда. В противном случае, New Logic имеет право отменить бронирование на условиях незаезда (см. Условия аннуляции).</BookingPolicyInfo>
          <CancellationPolicyInfo>Заказ может быть аннулирован бесплатно до времени наступления штрафных санкций. В случае поздней аннуляции или незаезда гостя система бронирования взимает 100% стоимости номера за 1 сутки.</CancellationPolicyInfo>
        </Room>
        <Room id="1219">
          <RoomActionTariff id="0" name=""/>
          <RoomType id="9" name="DBL/TWN"/>
          <Placement id="7"/>
          <FoodType id="19"/>
          <Price>6843</Price>
          <PricePerDay>
            <PricePerDayDetails>
              <Date>2014-08-13</Date>

```

```
<Price>2281</Price>
<Access>Номера доступны для моментального бронирования</Access>
<Allotment>5</Allotment>
</PricePerDayDetails>
<PricePerDayDetails>
  <Date>2014-08-14</Date>
  <Price>2281</Price>
  <Access>Номера доступны для моментального бронирования</Access>
  <Allotment>1</Allotment>
</PricePerDayDetails>
<PricePerDayDetails>
  <Date>2014-08-15</Date>
  <Price>2281</Price>
  <Access>Номера доступны для моментального бронирования</Access>
  <Allotment>1</Allotment>
</PricePerDayDetails>
</PricePerDay>
<Access>Номера доступны для моментального бронирования</Access>
<Allotment>1</Allotment>
<CancellationPolicy>
  <Price>2281</Price>
  <DateTimeFrom>2014-08-10 14:00:00</DateTimeFrom>
</CancellationPolicy>
<BookingPolicyInfo>Бронирование является гарантированным до 18:00 в день заезда. Пребывающие
после 18:00 гости должны уведомить об этом в поле «Примечания» во время бронирования или сообщением по
электронной почте, с указанием номера бронирования до 14:00 дня заезда. В противном случае, New Logic имеет
право отменить бронирование на условиях незаезда (см. Условия аннуляции).</BookingPolicyInfo>
<CancellationPolicyInfo>Заказ может быть аннулирован бесплатно до времени наступления штрафных
санкций. В случае поздней аннуляции или незаезда гостя система бронирования взимает 100% стоимости номера
за 1 сутки.</CancellationPolicyInfo>
</Room>
</Hotel>
</SearchHotelRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

3.2 Getting prices for early check-in and late check-out

Description:

Getting prices for early check-in and late check-out. Using this method is not necessary. It can be used if necessary to book early check-in and/or late check-out to show value fee for early check-in/late check-out. This method does not affect the process of booking and is informative.

Prices for early check-in/late check-out may be different at different times – for example, price for late check-out at 16:00 – 15 USD, and at 18:00 – 20 USD.

There are two types of early check-in – non-guaranteed and guaranteed. Their prices may be different, so the response given both types of early check-in.

Request attributes:

- lang – language for getting data (the parameter is optional)

Request elements:

- Room – room identifier

- RoomActionTariff – room action tariff identifier (if you need to book room by action tariff)
- Placement – placement type identifier
- FoodType – meals type identifier
- Currency – currency identifier
- Date – period of stay. Attributes:
 - from – check-in date
 - to – check-out date
- EarlyCheckIn – early check-in. Attributes (optional):
 - Time – if you need to get price of early check-in at the exact time
 - TimeFrom – if you need to get prices of early check-in for the period "from" a specified time
 - TimeTo – if you need to get prices of early check-in for the period "to" specified time

Note: if set none of the attributes, all prices of early check-in are presented in the response
- LateCheckOut – late check-out. Attributes (optional):
 - Time – if you need to get price of late check-out at the exact time
 - TimeFrom – if you need to get prices of late check-out for the period "from" a specified time
 - TimeTo – if you need to get prices of late check-out for the period "to" specified time

Note: if set none of the attributes, all prices of late check-out are presented in the response

As a result arrives the data packet with prices at the following structure. Response attributes and elements:

- Overtime – container for overtime price; contains price and period of time during which this price is operates
- Type – type of overtime. Available values:
 - Nonguaranteed early check-in
 - Guaranteed early check-in
 - Late check-out
- TimeFrom – the time, from which this price is operates
- TimeTo – the time, to which this price is operates
- Price – price for overtime

Note: prices for early check-in and late check-out can vary on different days. In the response are presented prices for a particular booking period specified in the request parameter Date.

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
```

```
<GetOvertimeRQ lang="en">
  <Auth login="test@test.com" pass="123"/>
  <Room>298</Room>
  <RoomActionTariff>2094</RoomActionTariff>
  <Placement>7</Placement>
  <FoodType>15</FoodType>
  <Currency id="17"/>
  <Date from="2015-08-25" to="2015-08-31"/>
  <EarlyCheckIn TimeFrom="09:00" TimeTo="13:00"/>
  <LateCheckOut Time="14:10"/>
</GetOvertimeRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetOvertimeRS>
      <Overtime>
        <Type>Late check-out</Type>
        <TimeFrom>14:01:00</TimeFrom>
        <TimeTo>15:00:00</TimeTo>
        <Price>408</Price>
      </Overtime>
      <Overtime>
        <Type>Nonguaranteed early check-in</Type>
        <TimeFrom>08:00:00</TimeFrom>
        <TimeTo>09:59:00</TimeTo>
        <Price>426</Price>
      </Overtime>
      <Overtime>
        <Type>Nonguaranteed early check-in</Type>
        <TimeFrom>10:00:00</TimeFrom>
        <TimeTo>11:59:00</TimeTo>
        <Price>535</Price>
      </Overtime>
      <Overtime>
        <Type>Nonguaranteed early check-in</Type>
        <TimeFrom>12:00:00</TimeFrom>
        <TimeTo>13:59:00</TimeTo>
        <Price>562</Price>
      </Overtime>
      <Overtime>
        <Type>Guaranteed early check-in</Type>
        <TimeFrom>00:00:00</TimeFrom>
        <TimeTo>13:59:00</TimeTo>
        <Price>719</Price>
      </Overtime>
    </GetOvertimeRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

3.3 Getting the list of hotel surcharges

Description:

For some hotels there may be surcharges that apply when ordering these hotels. Surcharges may be several for one hotel. All of them are added to the order. This method is designed to obtain a list of hotel surcharges.

For example, if the cost of living in a hotel is 200 USD and the hotel has a surcharge for booking 20 USD and an extra charge for tourist tax of 30 USD, then the total amount of the reservation will be 250 USD.

Request attributes:

- lang – language for getting data (the parameter is optional)

Request elements:

- Hotel – hotel identifier
- Currency – currency identifier

As a result arrives the data packet with surcharges at the following structure. Response attributes and elements:

- HotelSurcharge – hotel surcharge. Attributes:
 - id – surcharge identifier
 - name – surcharge name on the specified language
 - price – surcharge price in the specified currency

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetHotelSurchargesRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
      <Hotel id="80"/>
      <Currency id="17"/>
    </GetHotelSurchargesRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetHotelSurchargesRS>
      <HotelSurcharge id="1" name="Booking fee" price="350"/>
      <HotelSurcharge id="2" name="Tourist tax" price="500"/>
      <HotelSurcharge id="3" name="Services" price="200"/>
    </GetHotelSurchargesRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

3.4 Search of excursions

Description:

Search of excursions on the set parameters. Request attributes:

- lang – language for getting data (the parameter is optional)

Request elements (search parameters):

- City – city identifier
- Currency – currency identifier
- Date – period of holding the excursion. Attributes:
 - from – date begin of the excursion
 - to – date end of the excursion
- Adult – count of people for whom it will be holding the excursion

As a result of the search arrives the data packet at the following structure. Response attributes and elements:

- Excursion – container excursion. Attributes:
 - id – excursion identifier
- Price – price for excursion. Currency of price is currency that has been specified in the request parameters
- DateExcursion – date of excursion (for specification, if search was made for period of more than one day)
- CountPeopleMax – maximum possible count of people, for which can be holding the excursion
- CancellationPolicy – policy penalties on cancellation. Elements:
 - Price – penalty price
 - DateTimeFrom – date of penalty begin
- BookingPolicyInfo – information about reservation rules on the specified language
- CancellationPolicyInfo – information about cancellation rules on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchExcursionRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
      <City id="29"/>
      <Currency id="18"/>
      <Date from="2012-12-29" to="2012-12-30"/>
      <Adult count="1"/>
    </SearchExcursionRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchExcursionRS>
      <Excursion id="33">
        <Price>96</Price>
        <DateExcursion>2012-12-29</DateExcursion>
        <CountPeopleMax>3</CountPeopleMax>
        <CancellationPolicy>
          <Price>96</Price>
          <DateTimeFrom>2012-12-28 14:00:00</DateTimeFrom>
        </CancellationPolicy>
        <BookingPolicyInfo>Rule of booking of excursions</BookingPolicyInfo>
        <CancellationPolicyInfo>Rule of cancel of excursions</CancellationPolicyInfo>
      </Excursion>
    </SearchExcursionRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

3.5 Search of transfers

Description:

Search of transfers on the set parameters. Request attributes:

- lang – language for getting data (the parameter is optional)

Request elements (search parameters):

- City – city identifier
- Currency – currency identifier
- Date – period of holding the transfer
- Adult – count of people for whom it will be holding the transfer

As a result of the search arrives the data packet at the following structure. Response attributes and elements:

- Transfer – container transfer. Attributes:
 - id – transfer identifier
- Price – price for transfer. Currency of price is currency that has been specified in the request parameters
- CancellationPolicy – policy penalties on cancellation. Elements:
 - Price – penalty price
 - DateTimeFrom – date of penalty begin
- BookingPolicyInfo – information about reservation rules on the specified language
- CancellationPolicyInfo – information about cancellation rules on the specified language

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchTransferRQ lang="en">
```

```
<Auth login="test@test.com" pass="123"/>
<City id="29"/>
<Currency id="18"/>
<Date>2012-12-30</Date>
<Adult count="1"/>
</SearchTransferRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchTransferRS>
      <Transfer id="32">
        <Price>45</Price>
        <CancellationPolicy>
          <Price>45</Price>
          <DateTimeFrom>2012-12-28 14:00:00</DateTimeFrom>
        </CancellationPolicy>
        <BookingPolicyInfo>Rule of booking of excursions</BookingPolicyInfo>
        <CancellationPolicyInfo>Rule of cancel of excursions</CancellationPolicyInfo>
      </Transfer>
    </SearchTransferRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4 Operations with orders

4.1 Ordering

Description:

This paragraph describes the process of making new general order for the customer. General order is order which may consist of several unit orders: hotels, tours, transfers.

Ordering begins with authorization via API. In case of successful authorization checks the input parameters to create order. If authentication fails, the response is sent about the error authorization.

To booking the hotel uses the following parameters:

- hotel id
- period of stay (check-in date and check-out date)
- hotel room id
- type of meals
- type of placement
- count of hotel rooms, which includes the names of guests (for child is also required parameter is the age of the child), additional parameters: early check-in, type of early check-in, late check-out, additional wishes

Note: since there are two types of early check-in (non-guaranteed and guaranteed), when ordering early check-in you need to specify its type (parameter `TypeEarlyCheckIn`); available values:

- 1 – non-guaranteed early check-in
- 2 – guaranteed early check-in

If in the request is not specified parameter `TypeEarlyCheckIn`, by default ordering non-guaranteed early check-in.

To booking the excursion uses the following parameters:

- excursion id
- period of holding the excursion
- count of people participating in the excursion
- the start time of the excursion
- the meeting place
- additional wishes (optional)

To booking the transfer uses the following parameters:

- transfer id
- period of time for the transfer
- count of people to whom the transfer is offered
- the start time of the transfer
- the meeting place
- the delivery place
- additional wishes (optional)

Upon successful order creation comes the response with the following elements and attributes:

- Order – container order. Has attributes:

- UniqueID – unique order identifier
- Type – order type. Type «Main» indicates that it is a general order

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <AddOrderRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
      <Currency>18</Currency>
      <Hotel id="80">
        <Date from="2012-12-29" to="2012-12-30"/>
        <Room id="31">
          <RoomActionTariff>3</RoomActionTariff>
          <Placement>7</Placement>
          <FoodType>15</FoodType>
          <Amount>
            <Adult name="Test1"/>
            <Child name="Test2" age="4"/>
            <Option TimeEarly="09:00" TypeEarlyCheckIn="2" TimeDeparture="13:00" ClientHotelservice="Some client
note"/>
          </Amount>
          <Amount>
            <Adult name="Test3"/>
            <Child name="Test4" age="2"/>
          </Amount>
        </Room>
        <Room id="36">
          <RoomActionTariff>0</RoomActionTariff>
          <Placement>8</Placement>
          <FoodType>15</FoodType>
          <Amount>
            <Adult name="Test5"/>
          </Amount>
        </Room>
      </Hotel>
      <Excursion id="33">
        <Date from="2012-12-29" to="2012-12-29"/>
        <Persons>3</Persons>
        <Time>07:15</Time>
        <MettingPlace>Airport</MettingPlace>
        <Note>Some text</Note>
      </Excursion>
      <Transfer id="38">
        <Date from="2012-12-29" to="2012-12-29"/>
        <Persons>3</Persons>
        <Time>10:30</Time>
        <MettingPlace>Airport</MettingPlace>
        <DeliveryPlace>Dnepr</DeliveryPlace>
        <Note>Some text</Note>
      </Transfer>
    </AddOrderRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The above example shows how to issue one general order, consisting of three unit orders: hotel, excursion and transfer.

Unit order of hotel (id – 80) consist of three rooms (id – 31 (2 pcs.), id – 36) on period from 29.12.2012 to 30.12.2012.

First two rooms (id – 31) have: action tariff – 3, type of placement – 7, type of meals – 15. In the first of them will be living one adult, whose name is «Test1», and one child, whose name is «Test2» and age 4 years; this room has the following additional options: «TimeEarly» – early check-in at 09:00, «TypeEarlyCheckIn» – indicates that ordered guaranteed early check-in (value «2»), «TimeDeparture» – late check-out at 13:00, «ClientHotelservice» – the client wishes «Some client note». In the second room will be living one adult, whose name is «Test3», and one child, whose name is «Test4» and age 2 years; early check-in, late check-out and client wishes for this room are not set.

The third room (id – 36) has: action tariff – 0 (no action tariff), type of placement – 8, type of meals – 15. In this room will be living one adult, whose name is «Test5».

Unit order excursion (id – 33) will be ordered on 29.12.2012 for three people whom will meet at 07:15 in «Airport» and will take into account wish «Some text».

Unit order transfer (id – 38) will be ordered on 29.12.2012 for three people whom will meet at 10:30 in «Airport» and will taken to «Dnepr», also take into account wish «Some text».

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <AddOrderRS>
      <Order UniqueID="4G02YS" Type="Main"/>
    </AddOrderRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.2 Search of orders

Description:

Search of orders can be made on the check-in date, check-out date and hotel. Request elements (search parameters):

- DateFrom – start date of the period (optional parameter). If in request this parameter is absent, then as the value of this parameter is substituted current date
- DateTo – end date of the period (optional parameter). If in request this parameter is absent, then in result will be all orders that satisfy the previous parameter
- Hotel – hotel identifier (optional parameter). This parameter must be present in request, if it is necessary to obtain a list of orders for a specific hotel

As a result, comes the answer, that contains a list of orders, in which check-in/check-out date is in period from «DateFrom» to «DateTo», ordered in the hotel «Hotel». Response attributes and elements:

- Order – container order. Attributes:
 - UniqueID – unique order identifier
- OrderStatus – status of the order
- DateUpdate – date of last modification

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchOrdersRQ lang="ru">
      <Auth login="test@test.com" pass="123"/>
      <DateFrom date="2015-07-01"/>
      <DateTo date="2015-07-16"/>
    </SearchOrdersRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SearchOrdersRS>
      <Order UniqueID="WLAWWV">
        <OrderStatus>5</OrderStatus>
        <DateUpdate>2015-04-08</DateUpdate>
      </Order>
      <Order UniqueID="CDN34N">
        <OrderStatus>1</OrderStatus>
        <DateUpdate>2015-04-09</DateUpdate>
      </Order>
      <Order UniqueID="NIT2DF">
        <OrderStatus>5</OrderStatus>
        <DateUpdate>2015-04-24</DateUpdate>
      </Order>
    </SearchOrdersRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.3 Getting information about the order

Description:

Getting information about the order made by the following parameters:

- Order – container order. Attributes:
 - UniqueID – unique order identifier (general or unit)
 - Type – type of order. Valid values:
 - Main – general order
 - Hotel – unit order hotel
 - Excursion – unit order excursion
 - Transfer – unit order transfer

As a result, comes complete information about the order. Response attributes and elements:

- GeneralOrder – container of general order. Includes a plurality of containers OrderHotels, OrderExcursions, OrderTransfers. Attributes:
 - currency – order currency
 - UniqueID – identifier of general order
 - TotalPrice – total price of order
 - OrderStatus – status of general order
- OrderHotels – container of unit order hotel. Attributes:
 - UniqueID – identifier of unit order
 - PenaltyEdit – price of penalties for modification the order (*Note: price of penalties included in price of unit order, in the attribute Price*)
 - PenaltyCancel – price of penalties for cancellation the order (*Note: price of penalties included in price of unit order, in the attribute Price*)
- Hotel – container hotel. Attributes:
 - id – hotel identifier
- Date – period of stay in hotel/date of holding the excursion/date of transfer. Attributes:
 - from – period begin
 - to – period end
- Room – container hotel room. Attributes:
 - id – hotel room identifier
- RoomActionTariff – room action tariff. Attributes:
 - id – unique room action tariff identifier
 - name – room action tariff name on the specified language
- Price – price for hotel room/excursion/transfer
- Placement – placement type identifier
- FoodType – meals type identifier
- Amount – count of hotel rooms. It is determined from the count of residents
- Adult – adult. Attributes:
 - name – name of adult
- Child – child. Attributes:
 - name – name of child
 - age – age of child
- Option – additional parameters of hotel room. Attributes:
 - TimeEarly – early check-in
 - TimeDeparture – late check-out
 - ClientHotelservice – additional client wishes
- HotelSurcharge – hotel surcharge. Attributes:
 - id – surcharge identifier
 - name – surcharge name
 - price – surcharge price
- CancellationPolicy – policy penalties on cancellation. Elements:
 - Price – penalty price
 - DateTimeFrom – date of penalty begin
- OrderExcursions – container of unit order excursion. Attributes like in element OrderHotels

- Excursion – container excursion. Attributes:
 - id – excursion identifier
- Persons – count of people taking part in excursion/transfer
- Time – time of beginning for excursion/transfer
- MeetingPlace – meeting place for excursion/transfer
- Note – note, wish for excursion/transfer
- Transfer – container transfer. Attributes:
 - id – transfer identifier
- DeliveryPlace – delivery place for transfer

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetOrderInfoRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
      <Order UniqueID="RSA1R1" Type="Main"/>
    </GetOrderInfoRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetOrderInfoRS>
      <GeneralOrder currency="18" UniqueID="RSA1R1" TotalPrice="671" OrderStatus="1">
        <OrderHotels UniqueID="RSA1R1A" PenaltyEdit="20" PenaltyCancel="0">
          <Hotel id="80">
            <Date from="2012-12-29" to="2012-12-30"/>
            <Room id="31">
              <RoomActionTariff id="3" name="Non refundable tariff"/>
              <Price>470</Price>
              <Placement>7</Placement>
              <FoodType>15</FoodType>
              <Amount>
                <Adult name="test1"/>
                <Child name="test1" age="4"/>
                <Option TimeEarly="09:00:00" TimeDeparture="23:00:00" ClientHotelservice="Some client note"/>
              </Amount>
              <Amount>
                <Adult name="test2"/>
                <Child name="test2" age="2"/>
                <Option/>
              </Amount>
            </Room>
            <HotelSurcharge id="1" name="Booking fee" price="50"/>
            <HotelSurcharge id="2" name="Tourist tax" price="20"/>
          </Hotel>
          <CancellationPolicy>
            <Price>151</Price>
            <DateTimeFrom>2012-12-28 14:00:00</DateTimeFrom>
```

```

    </CancellationPolicy>
</OrderHotels>
<OrderExcursions UniqueID="RSA1R1C" PenaltyEdit="10" PenaltyCancel="0">
  <Excursion id="33">
    <Price>96</Price>
    <Date from="2012-12-29" to="2012-12-29"/>
    <Persons>3</Persons>
    <Time>07:15</Time>
    <MettingPlace>Airport</MettingPlace>
    <Note>some text1</Note>
  </Excursion>
  <CancellationPolicy>
    <Price>96</Price>
    <DateTimeFrom>2012-12-28 14:00:00</DateTimeFrom>
  </CancellationPolicy>
</OrderExcursions>
<OrderTransfers UniqueID="RSA1R1D">
  <Transfer id="38">
    <Price>35</Price>
    <Date from="2012-12-29" to="2012-12-29"/>
    <Persons>3</Persons>
    <Time>10:30</Time>
    <MettingPlace>Airport</MettingPlace>
    <DeliveryPlace>Dnepr</DeliveryPlace>
    <Note>some text2</Note>
  </Transfer>
  <CancellationPolicy>
    <Price>35</Price>
    <DateTimeFrom>2012-12-28 14:00:00</DateTimeFrom>
  </CancellationPolicy>
</OrderTransfers>
</GeneralOrder>
</GetOrderInfoRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.4 Order modification

Description:

Modification of order takes place by set of new parameters for the order. Upon successful verification of all parameters and modification capability on the new parameters it updates the order, identifier of which is passed in the parameter UniqueID. Request attributes and elements:

- Currency – order currency
- UniqueID – unique identifier of general order
- Hotel – container of unit order hotel. Attributes:
 - UniqueID – unique identifier of unit order
 - id – hotel identifier
- Date – period of unit order. Attributes:
 - from – period begin
 - to – period end
- Room – container hotel room. Attributes:
 - id – hotel room identifier
- RoomActionTariff – room action tariff

- Placement – placement type identifier
- FoodType – meals type identifier
- Amount – count of hotel rooms. It is determined from the count of residents
- Adult – adult. Attributes:
 - name – name of adult
- Child – child. Attributes:
 - name – name of child
 - age – age of child
- Option – additional parameters of hotel room. Attributes:
 - TimeEarly – early check-in
 - TypeEarlyCheckIn – type of early check-in. Available values:
 - 1 – non-guaranteed early check-in
 - 2 – guaranteed early check-in
 - TimeDeparture – late check-out
 - ClientHotelservice – additional client wishes

Attributes for modification order excursion and transfer are similar to the parameters of the method "[Ordering](#)".

Upon successful order modification comes the answer with the following elements and attributes:

- Order – container order. Has attributes:
 - UniqueID – unique order identifier
 - Type – order type. Type «Main» indicates that it is a general order

Request example:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <EditOrderRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
      <Currency>18</Currency>
      <UniqueID>RSA1R1</UniqueID>
      <Hotel UniqueID="RSA1R1A" id="80">
        <Date from="2012-12-29" to="2012-12-30"/>
        <Room id="31">
          <RoomActionTariff>3</RoomActionTariff>
          <Placement>7</Placement>
          <FoodType>15</FoodType>
          <Amount>
            <Adult name="test1"/>
            <Child name="test2" age="4"/>
            <Option TimeEarly="09:00" TypeEarlyCheckIn="1" TimeDeparture="23:00" ClientHotelservice="Some client
note"/>
          </Amount>
          <Amount>
            <Adult name="test3"/>
            <Child name="test4" age="2"/>
            <Option/>
          </Amount>
        </Room>
      </Hotel>
    </EditOrderRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

```

</Hotel>
<Excursion UniqueID="RSA1R1C" id="33">
  <Date from="2012-12-29" to="2012-12-29"/>
  <Persons>3</Persons>
  <Time>07:15</Time>
  <MettingPlace>Airport</MettingPlace>
  <Note>some text</Note>
</Excursion>
<Transfer UniqueID="RSA1R1D" id="38">
  <Date from="2012-12-29" to="2012-12-29"/>
  <Persons>3</Persons>
  <Time>10:30</Time>
  <MettingPlace>Airport</MettingPlace>
  <DeliveryPlace>Dnepr</DeliveryPlace>
  <Note>some text</Note>
</Transfer>
</EditOrderRQ>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Response example:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <EditOrderRS>
      <Order UniqueID="4G02YS" Type="Main"/>
    </EditOrderRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

4.5 Simulating order modification

Description:

This method is designed to simulate order modification in order to see what it will look like after modification. It is not necessary to use this method. It is intended for informative purpose only.

Attention! This method *does not* edit the order, but only simulates it. To edit an order, you must use the "[Order modification](#)" method. Between the execution of the SimulateEditOrder method and the execution of the EditOrder method, the information on the server may change. Therefore, to view the information of the modified order after the "[Order modification](#)" method is executed, you need to use the "[Getting information about the order](#)" method.

Note: simulating order modification works only for hotel orders. For excursion orders and transfer orders this method does not work.

Request attributes and elements in this method are the same as in the request of the "[Order modification](#)" method. Response attributes and elements in this method are the same as in the response of the "[Getting information about the order](#)" method.

Request example:

```

<?xml version="1.0" encoding="UTF-8"?>

```

```

<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SimulateEditOrderRQ lang="en">
      <Auth login="test@test.com" pass="123"/>
      <Currency>18</Currency>
      <UniqueID>RSA1R1</UniqueID>
      <Hotel UniqueID="RSA1R1A" id="80">
        <Date from="2012-12-29" to="2012-12-30"/>
        <Room id="31">
          <RoomActionTariff>3</RoomActionTariff>
          <Placement>7</Placement>
          <FoodType>15</FoodType>
          <Amount>
            <Adult name="test1"/>
            <Child name="test2" age="4"/>
            <Option TimeEarly="09:00" TypeEarlyCheckIn="1" TimeDeparture="23:00" ClientHotelservice="Some client
note"/>
          </Amount>
          <Amount>
            <Adult name="test3"/>
            <Child name="test4" age="2"/>
            <Option/>
          </Amount>
        </Room>
      </Hotel>
    </SimulateEditOrderRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Response example:

```

<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SimulateEditOrderRS>
      <GeneralOrder currency="18" UniqueID="RSA1R1" TotalPrice="540" OrderStatus="9">
        <OrderHotels UniqueID="RSA1R1A" PenaltyEdit="20" PenaltyCancel="0">
          <Hotel id="80">
            <Date from="2012-12-29" to="2012-12-30"/>
            <Room id="31">
              <RoomActionTariff id="3" name="Non refundable tariff"/>
              <Price>470</Price>
              <Placement>7</Placement>
              <FoodType>15</FoodType>
              <Amount>
                <Adult name="test1"/>
                <Child name="test2" age="4"/>
                <Option TimeEarly="09:00:00" TimeDeparture="23:00:00" ClientHotelservice="Some client note"/>
              </Amount>
              <Amount>
                <Adult name="test3"/>
                <Child name="test4" age="2"/>
                <Option/>
              </Amount>
            </Room>
            <HotelSurcharge id="1" name="Booking fee" price="50"/>
            <HotelSurcharge id="2" name="Tourist tax" price="20"/>
          </Hotel>

```

```
<CancellationPolicy>
  <Price>151</Price>
  <DateTimeFrom>2012-12-28 14:00:00</DateTimeFrom>
</CancellationPolicy>
</OrderHotels>
</GeneralOrder>
</SimulateEditOrderRS>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.6 Order confirmation

Description:

Orders, that are available for instant booking (online), confirmed immediately at the time of booking, thus further confirming its not need. This method applies only to orders that are available on request (onrequest).

When booking order, available on request (onrequest), a request is sent to the supplier for availability. This order meanwhile transferred to the status «requested». The supplier can confirm the order (in case of availability), or cancel the order (in case of not availability). If the provider confirms the order, it is transferred to the status «confirmed». In this case, the client will have to confirm his booking. This is why this method is needed.

There are two types of order – general and unit. General order consists of a plurality of unit orders. It is possible to confirm any number of general or unit order. In case of confirmation all single orders included into general one, general order is confirmed automatically. In case of confirmation general order, all single orders included into it are also confirmed automatically (only those that is in status «confirmed»). If it is necessary to confirm all unit orders, included into general one, it will be enough to send request for the confirmation of general order.

Note: you can confirm only orders in status «confirmed». If in request were set orders in other statuses – such orders will be ignored.

For the confirmation of any order, you need to send request that will contain the order identifier (UniqueID) and its type (Type).

Response will contain a list of all confirmed orders and their types. If in request was made a mistake in the order identifier or was sent identifier of the order, which is owned by another user, then the order will not be confirmed and will not be present in response.

Elements and attributes:

- Order – container order. Has attributes:
 - UniqueID – unique identifier of the order (general or unit)
 - Type – order type. Valid values:
 - Main – general order
 - Hotel – unit order hotel
 - Excursion – unit order excursion
 - Transfer – unit order transfer

Request example of confirmation two general orders and three unit orders (all orders are not linked):

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <ConfirmOrderRQ>
      <Auth login="test@test.com" pass="123"/>
      <Order UniqueID="SG69XF" Type="Main"/>
      <Order UniqueID="6SW8FGA" Type="Hotel"/>
      <Order UniqueID="6QECIY" Type="Main"/>
      <Order UniqueID="G3KW8KA" Type="Hotel"/>
      <Order UniqueID="6QECIYB" Type="Excursion"/>
    </ConfirmOrderRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example for previous request:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <ConfirmOrderRS>
      <Order UniqueID="SG69XF" Type="Main"/>
      <Order UniqueID="6SW8FGA" Type="Hotel"/>
      <Order UniqueID="6QECIY" Type="Main"/>
      <Order UniqueID="G3KW8KA" Type="Hotel"/>
      <Order UniqueID="6QECIYB" Type="Excursion"/>
    </ConfirmOrderRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

4.7 Order cancellation

Description:

There are two types of order – general and unit. General order consists of a plurality of unit orders. It is possible to cancel any number of general or unit order. In case of cancellation all single orders included into general one, general order is cancelled automatically. In case of cancellation general order, all single orders included into it are also cancelled automatically. If it is necessary to cancel all unit orders, included into general one, it will be enough to send request for the cancellation of general order.

Note: you can cancel orders in any status.

For the cancellation of any order, you need to send request that will contain the order identifier (UniqueID) and its type (Type).

Response will contain a list of all canceled orders and their types. If in request was made a mistake in the order identifier or was sent identifier of the order, which is owned by another user, then the order will not be canceled and will not be present in response.

Elements and attributes:

- Order – container order. Has attributes:

- UniqueID – unique identifier of the order (general or unit)
- Type – order type. Valid values:
 - Main – general order
 - Hotel – unit order hotel
 - Excursion – unit order excursion
 - Transfer – unit order transfer

Request example of cancellation two general orders and three unit orders (all orders are not linked):

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CancelOrderRQ>
      <Auth login="test@test.com" pass="123"/>
      <Order UniqueID="SG69XF" Type="Main"/>
      <Order UniqueID="6SW8FGA" Type="Hotel"/>
      <Order UniqueID="6QECIY" Type="Main"/>
      <Order UniqueID="G3KW8KA" Type="Hotel"/>
      <Order UniqueID="6QECIYB" Type="Excursion"/>
    </CancelOrderRQ>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Response example for previous request:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <CancelOrderRS>
      <Order UniqueID="SG69XF" Type="Main"/>
      <Order UniqueID="6SW8FGA" Type="Hotel"/>
      <Order UniqueID="6QECIY" Type="Main"/>
      <Order UniqueID="G3KW8KA" Type="Hotel"/>
      <Order UniqueID="6QECIYB" Type="Excursion"/>
    </CancelOrderRS>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

5 Contacts

If you have any questions, you can ask them the following contacts:

Organizational issues:

Sergiy Varfolomeev – head of reservation systems dept., New Logic

sv@newlogic.ua

+38-044-206-33-22

Technical issues:

Oleksandr Trelin – technical specialist of the developer company, TreoLabs GmbH

o.trelin@treolabs.com