

Travel Search Web Services

API to search, book, modify and cancel booking of packages, hotels, flights and travel services

Index

Introduction	2
Authorization.....	2
Links for Web services.....	2
XML messages overview	2
Operations description.....	3
Booking workflow description	7
Search Tour product.....	8
Flight availability requests.....	8
Standard package availability request	16
Getting Options	21
Book Order workflow	26
E-tickets issue	31
Booking modifications.....	32
Booking cancellation	32

Introduction

Online Travel web services provide software developers with a possibility to create integration with the Company's Information System to search and book travel packages. Developers seeking to use these services should have a background in web services development.

Authorization

To connect to the Information System using web services, customers will need valid user credentials (login and password).

Links for Web services

1. Web services for Tour Products Search requests, Travel guide references (Countries, Regions, Cities, Air companies etc.) requests:

http://***/webservice/SearchTourWS

Description of these web services can be found at http://***/webservice/SearchTourWS?WSDL

2. Web services for Booking/Cancellation/Update requests and getting some additional information:

http://***/webservice/BookingWS

Description of these web services can be found at http://***/webservice/BookingWS?WSDL

3. More web services for Travel guide references, partly duplicated with *SearchTourWS* (Packages' templates, Countries, Regions, Cities, Hotels, Boards Types, and Accommodation types' lists requests by parent ID):

http://***/webservice/GeoService

Description of these web services: http://***/webservice/GeoService?WSDL

XML messages overview

To use our web services you have to prepare the request and handle response. All requests and responses have similar structure. Every request consists of *WsLoginPassword* object (login/password and *AgencyUnit* ID, *idContract* id) and the specific petitions. For example: *OrderInfoRequest* element has only one attribute "orderId".

The Response consists of two parts "*ResponseLog*" and "*XXXSpecificResponseData*". "*ResponseLog*" attribute "comment" contains error message text or "comment" is empty if request/response is successful. First of all you have to handle the "*ResponseLog*".

Operations description

Terms

- *Order* – an entity in the Information System which can contain of one or more products requested by one customer (agency).
- *Tour Product* (Product) – an entity in the Information System based on a fixed Set of Services (so called Template). Each Service in the Product can be related to its own passenger(s). Passengers of an Order are the join of all products' passengers.
- *Service* – a basic entity in the Information System which describes tourist services. The types of services are: accommodation, cruise, roundtrip, flight, transfers (2 types - regular and individual), insurance, excursion, visa, and services called "miscellaneous". Services cannot be added to Orders directly - only as a part of a Tour Product!
- *Passenger* – consumer of a service.

Table: Available operations (alphabetical, by WS Operation name)

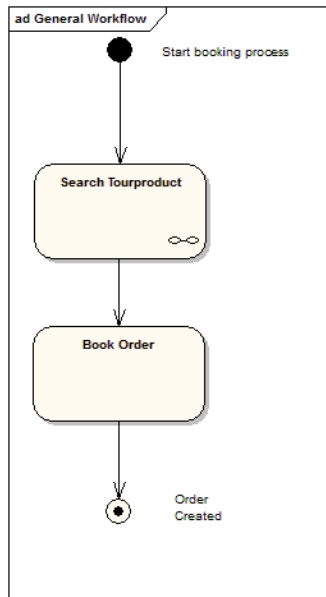
No	WS Operation name	Operation	Operation Type	Operation description	Web service name
1	<i>addTourproduct</i>	Add Tour product to an existing Order	Reservation	This operation allows to add a new Tour product to the existing Order	<i>BookingWS</i>
2	<i>bookOrder</i>	Order reservation Request	Reservation	First operation for Order creation. Agency unit and contract are needed for Order creation. List of unit and contracts you can get by operations <i>getAgencyContracts</i> and <i>getAgencyUnits</i> .	<i>BookingWS</i>
3	<i>cancelOneTourProduct</i>	Cancel one package from an Order	Cancellation	Cancel a Service of the Order (created by means of <i>bookOrder</i>). You can get information about services of the Order by means of "getOrderInfo" operation	<i>BookingWS</i>
4	<i>cancelOrder</i>	Cancel an Order	Cancellation	Cancel existing Order (created by means of <i>bookOrder</i>).	<i>BookingWS</i>
5	<i>changeTourproduct</i>	Modify (change) Tourproduct	Reservation	Keeping the Booking number	<i>BookingWS</i>
6	<i>getAgencyContracts</i>	Agency Contract List Request	Get a List	Gets a list of Contracts of the Agency logged in: information about a contract is mandatory for Order creation	<i>BookingWS</i>
7	<i>getAgencyUnits</i>	Agency Offices List Request	Get a List	Gets a list of Offices of the Agency logged in: information about a contract is mandatory for Order creation	<i>BookingWS</i>
8	<i>getAirAccommodation</i>	Flights availability request	Get a List	Returns flights availability for a period, e.g., to show on the website	<i>BookingWS</i>
9	<i>getAirCompanies</i>	Air Companies list Request	Get a List	Allows to get a list of Air Companies available	<i>BookingWS, SearchTourWS</i>
10	<i>getAirportByID</i>	Airports list Request	Get a List	Allows to get a list of Airports available	<i>BookingWS, SearchTourWS</i>

11	<i>getAllAccomdTypes</i>	Accommodation Types list Request	Get a List	This operation is used to get a list of Accommodation Types available. An Accommodation Type contents an information about types (Adult, Child) and numbers of customers allowed for this particular accommodation, e.g. "DBL + CHD (2-4) + CHD (5-12)"	<i>BookingWS, SearchTourWS</i>
12	<i>getAllAgeTypes</i>	Age Types list Request	Get a List	Allows to get a list of Age Types available. An Age Type contents an information about a type (Adult, Child) and age range allowed for this type, to be used in an Accommodation Type, e.g. "CHD(2-12)"	<i>BookingWS, SearchTourWS</i>
13	<i>getAllAirportCities</i>	Cities of Airports list Request	Get a List	Returns a list of Airport cities	<i>BookingWS, SearchTourWS</i>
14	<i>getAllAirports</i>	Airport list request	Get a List	Returns a list of Airports	<i>SearchTourWS</i>
15	<i>getAllBoardTypes</i>	Meal Board Types list Request	Get a List	This operation allows to get a list of Meal Board Types	<i>BookingWS, SearchTourWS</i>
16	<i>getAllCities</i>	Cities list Request	Get a List	This operation allows to get a list of Cities (Fixed tree structure is used: City -> Region -> Country)	<i>BookingWS, SearchTourWS</i>
17	<i>getAllClassOfSeats</i>	Flight classes list request	Get a List	A (business), C (business), U (economy), N (economy) etc.	<i>BookingWS, SearchTourWS</i>
18	<i>getAllClassOfSeatsGroups</i>	Class of seats list request	Get a List	First, Business, Economy, Premium etc.	<i>BookingWS, SearchTourWS</i>
19	<i>getAllCountries</i>	Countries list Request	Get a List	Allows to get a list of Countries available	<i>BookingWS, SearchTourWS</i>
20	<i>getAllCruiseCompanies</i>	Cruise Companies list Request	Get a List	Allows to get a list of Cruise Companies names	<i>BookingWS, SearchTourWS</i>
21	<i>getAllCruiseNames</i>	Cruise Names list Request	Get a List	Allows to get a list of Cruise (that is, Cruise Routs) names, like "European Grand Voyage" etc. A Cruise is combining an Itinerary, and a Cruise ship info.	<i>BookingWS, SearchTourWS</i>
22	<i>getAllCurrencies</i>	Currencies list Request	Get a List	Allows to get a list of Currencies in which prices can be represented	<i>BookingWS, SearchTourWS</i>
23	<i>getAllRegions</i>	Regions list Request	Get a List	This operation allows to get a list of Regions (Fixed tree structure is used: City -> Region -> Country)	<i>BookingWS, SearchTourWS</i>
24	<i>getAllRoomTypes</i>	Room Types list Request	Get a List	Allows to get a full list of Room Types common for all hotels (e.g. Standard, Suite, Family etc.)	<i>BookingWS, SearchTourWS</i>
25	<i>getAllShips</i>	Ships list Request	Get a List	Allows to get a list of Cruise Ships used to describe a Cruise	<i>SearchTourWS</i>
26	<i>getAllTemplateAddons</i>	Extras request	Get a List	Returns a list of all extras for the particular Template	<i>BookingWS</i>

27	<i>getAllTemplateItems</i>	Get a list of all items included in the Template	Get a List	Gets a list of all items included in the Template	<i>BookingWS</i>
28	<i>getCarrierRentAddon</i>	Additional Rent a car services request	Get a List	Gets a list of additional rent a car services, for the main product	<i>BookingWS</i>
29	<i>getDemandAgentInfo</i>	List of Orders request	Get a List	Returns a list of Orders accessible for user	<i>BookingWS</i>
30	<i>getDemandView</i>	Order Info	Read Request	Order info request	<i>BookingWS</i>
31	<i>getFlightAddon</i>	Additional Flights request	Get a List	Gets a list of extra flights additional to the main product	<i>BookingWS</i>
32	<i>getFlightVariants</i>	Options request, for Flights	Get a List	Returns a list of options if Flights are variable	<i>BookingWS</i>
33	<i>getGroupExcursionAddon</i>	Extra Group Excursions request	Get a List	Gets a list of Group Excursions additionally to the main tour product	<i>BookingWS</i>
34	<i>getGroupExcursionVariants</i>	Options request, for Group Excursions	Get a List	Returns a list of options for Group Excursions if they are variable	<i>BookingWS</i>
35	<i>getGroupTransferAddon</i>	Additional Group transfers request	Get a List	Gets a list of additional Group transfers, for the main product	<i>BookingWS</i>
36	<i>getIndividualTransferAddon</i>	Additional Individual transfers request	Get a List	Gets a list of additional Individual transfers, for the main product	<i>BookingWS</i>
37	<i>getInsuranceAddon</i>	Additional Insurance request	Get a List	Gets a list of additional Insurance, for the main product	<i>BookingWS</i>
38	<i>getMiscCommodityAddon</i>	Additional services of "Miscellaneous" type request	Get a List	Gets a list of additional "Miscellaneous" services, for the main product	<i>BookingWS</i>
39	<i>getOrderInfo</i>	Read an Order Info	Read Request	Allows to get information about the Services (Tour Products) included in the Order and their passengers	<i>BookingWS</i>
40	<i>getResidenceAddon</i>	Additional Accommodations request	Get a List	Gets a list of extra accommodations additional to the main product	<i>BookingWS</i>
41	<i>getResidenceVariants</i>	Options request, for Accommodations	Get a List	Returns a list of options if Accommodation is variable	<i>BookingWS</i>
42	<i>getSightseeingVariants</i>	Roundtrip options request	Get a List	Returns a list of options for Roundtrip if it is variable	<i>BookingWS</i>

43	<i>getTariffTypes</i>	Tariff Type list Request	Get a List	Allows receiving a list of Tariff Types. A Tariff Type can be related to one of Tariff Kinds (Basic, Pronto, Last Minute, and Special) and is describing Booking and Payment conditions (Available booking dates, payment due dates) and Cancellation fees in percentage to the Service price	<i>BookingWS, SearchTourWS</i>
44	<i>getTourProductSetItems</i>	Get a list of all items included in the tour product	Get a List	Optional request	<i>BookingWS</i>
45	<i>getTourProductTariff</i>	Tour product items availability request	Availability Request	Returns availability for all items included in the tour product	<i>SearchTourWS</i>
46	<i>getVisaAddon</i>	Additional Visa services request	Get a List	Gets a list of additional visa services, for the main product	<i>BookingWS</i>
47	<i>searchCruise</i>	Cruises Availability Request	Availability Request	Makes a search of Products containing Cruises as Main Service.	<i>searchCruise</i>
48	<i>searchFlight</i>	Flights Availability	Availability Request	Makes a search of Products containing Flights as Main Service.	<i>searchCruise</i>
49	<i>searchFlightSpecial</i>	Flight availability request (special fares)	Availability Request	Special tourproducts and fares, for registred agencies	<i>SearchTourWS</i>
50	<i>searchRoundtrip</i>	Roundtrips Availability	Availability Request	Makes a search of Products containing Roundtrip as Main Service.	<i>BookingWS</i>
51	<i>searchStandardTour</i>	Packages Availability Request	Availability Request	Makes a search of Products containing Accommodation as Main Service.	<i>BookingWS</i>
52	<i>setAirtrip</i>	Set Final staus	Update	Modify final status of Flights ("ready to issue electronic airtickets"): true/false	<i>BookingWS</i>
53	<i>setAirtripReady</i>	Final status update	Update	Set final status to "ready to issue electronic airtickets"	<i>BookingWS</i>
54	<i>signTickets</i>	Issue airtickets	Update	Issue electronic tickets	<i>BookingWS</i>
55	<i>updatePassenger</i>	Passenger Data Update	Update	Updates Passenger personal data	<i>BookingWS</i>

Booking workflow description



Description:

- To create a booking, you have to do first a Tour product availability request, using one of the web services searching a package which has as a Main service one of the following services:
 - Accommodation (*searchStandardTour*),
 - Flight (*searchFlight* and *searchFlightSpecial*)
 - Roundtrip (*searchRoundtrip*),
 - Cruise (*searchCruise*).
- Get a response with price and availability;
- Create an Order adding Tour Product ID (only one) and Passengers information using bookOrder request;
- Retrieve the response with Order ID, Tour Product and its Passengers Info.

Important note: using Web Services, you can only book services like accommodation, flights, roundtrips etc. as a part of a package consisting of one or more items. One of package's items is defined as Main Service, to allow you to search by its attributes.

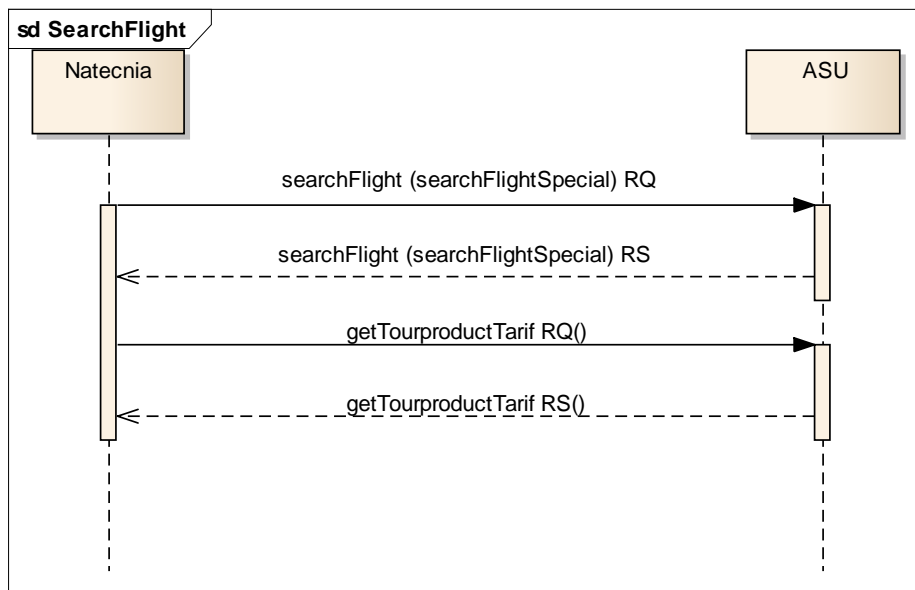
Search Tour product

Flight availability requests

There are 2 kind of flight availability requests: for public fares (*searchFlight*) offered on the website and accessible to all agencies and for special fares accessible to the particular agencies (*searchFlightSpecial*).

To use these requests, it is to know the *tourProductTempIID*: e.g. for searching standard One way flights, the *tourProductTempIID*=298; for Roundtrips, *tourProductTempIID*=216. **Please note:** to exclude inconveniences, it is always to specify *tourProductTempIID* for the particular integration.

The sequence diagram is:



SearchFlight

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***.web services"
xmlns:java="java:com.nt.slib.ws.wssearch" xmlns:java1="java:com.nt.slib.ws.wssearch.request">
  <soapenv:Header/>
  <soapenv:Body>
    <web:searchFlight>
      <web:wsLoginPassword>
        <java:wslogin>***</java:wslogin>
        <java:wspassword>***</java:wspassword>
      </web:wsLoginPassword>
      <web:flightRequest>
        <java1:ageTypeID>101</java1:ageTypeID>
        <java1:airCompanyCode>?</java1:airCompanyCode>
        <java1:airDateFromDestination>29.11.2014</java1:airDateFromDestination>
        <java1:airDateFromOrigin>22.11.2014</java1:airDateFromOrigin>
        <java1:airType>?</java1:airType>
        <java1:classofSeatsTypeID>?</java1:classofSeatsTypeID>
        <java1:destinationAirPortID>?</java1:destinationAirPortID>
        <java1:destinationCityID>1</java1:destinationCityID>
        <java1:flightNumber>331</java1:flightNumber>
        <java1:originAirPortID>?</java1:originAirPortID>
        <java1:originCityID>541</java1:originCityID>
        <java1:priceMax>?</java1:priceMax>
        <java1:priceMin>?</java1:priceMin>
        <java1:rowNumFrom>0</java1:rowNumFrom>
        <java1:rowNumTo>10</java1:rowNumTo>
      </web:flightRequest>
    </web:searchFlight>
  </soapenv:Body>
</soapenv:Envelope>
```

```
<java1:tariffTypeID?></java1:tariffTypeID>
<java1:tourProductTemplID>216</java1:tourProductTemplID>
</web:flightRequest>
</web:searchFlight>
</soapenv:Body>
</soapenv:Envelope>
```

Comments:

- *ageTypeID*: optional: you can get by *getAllAgeTypes* Request (e.g. 101 for Adults, 163 for Infants (age 0-1), 111 for Children (age 2-11))
- *airCompanyCode*: optional, can be received using *getAirCompanies* request
- *airDateFromDestination*: start date, mandatory
- *airDateFromOrigin*: end date, mandatory
- *airType*: optional: One Way (1) or Roundtrips (2). Optional: now only One Way flights are possible to book.
- *classOfSeatsTypeID*: (actually, Class Of Seats IDs – a mess found (?) optional, carrier's booking classes; can be received by *getAllClassOfSeats* request
- *destinationAirportID*: optional; for a particular city, can be received with *getAirportByID* request
- *destinationCityID*: mandatory; to get, please use *getAllAirportCities* request
- *flightNumber*: mandatory, from the list of flights agreed in advance
- *originAirportID*: optional; see *getAirportByID* request
- *originCityID*: mandatory, see *getAllAirportCities* request
- *priceMax*: optional
- *priceMin*: optional
- *rowNumFrom*: mandatory, first number of rows displayed in search results; counted from 0
- *rowNumTo*: mandatory, last number of rows displayed in search results; counted from 1
- *tariffTypeID*: tariff types of Tourproducts IDs, optional; see *getTariffTypes* request.
- *tourProductTemplID*: ID of Tourproduct Template, subject to specify.

Response:

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:searchFlightResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:flights xsi:type="n3:ArrayOfFlight" xmlns:n3="java:com.nt.slib.ws.wssearch.rows"
xmlns:n2="java:com.nt.slib.ws.wssearch.response">
          <n3:Flight>
            <n3:duration>7</n3:duration>
            <n3:flightSegments xsi:type="n3:ArrayOfFlightSegment">
              <n3:FlightSegment>
                <n3:ageType>ADL</n3:ageType>
                <n3:ageTypeID>101</n3:ageTypeID>
                <n3:aircompanyCode>UN</n3:aircompanyCode>
                <n3:classOfSeats>U</n3:classOfSeats>
                <n3:classOfSeatsID>2</n3:classOfSeatsID>
                <n3:destinationAirportCode>BCN2</n3:destinationAirportCode>
                <n3:destinationAirportID>839</n3:destinationAirportID>
                <n3:destinationCity>Барселона</n3:destinationCity>
                <n3:flightNumber>331</n3:flightNumber>
                <n3:offset>0</n3:offset>
                <n3:originAirportCode>DME</n3:originAirportCode>
                <n3:originAirportID>840</n3:originAirportID>
                <n3:originCity>Москва</n3:originCity>
                <n3:timetable>
                  <n3:airportFromCode>DME</n3:airportFromCode>
                  <n3:airportToCode>BCN2</n3:airportToCode>
                  <n3:arrivalTime>13:25</n3:arrivalTime>
                  <n3:dayShift>0</n3:dayShift>
                  <n3:departureTime>10:45</n3:departureTime>
                  <n3:ticketNumber xsi:nil="true"/>
                </n3:timetable>
              </n3:FlightSegment>
            </n3:FlightSegment>
          </n3:Flight>
        </n2:flights>
      </n1:result>
    </m:searchFlightResponse>
  </env:Body>
</env:Envelope>
```

```
<n3:ageType>ADL</n3:ageType>
<n3:ageTypeID>101</n3:ageTypeID>
<n3:aircompanyCode>UN</n3:aircompanyCode>
<n3:classOfSeats>U</n3:classOfSeats>
<n3:classOfSeatsID>2</n3:classOfSeatsID>
<n3:destinationAirportCode>DME</n3:destinationAirportCode>
<n3:destinationAirportID>840</n3:destinationAirportID>
<n3:destinationCity>Москва</n3:destinationCity>
<n3:flightNumber>332</n3:flightNumber>
<n3:offset>0</n3:offset>
<n3:originAirportCode>BCN2</n3:originAirportCode>
<n3:originAirportID>839</n3:originAirportID>
<n3:originCity>Барселона</n3:originCity>
<n3:timetable>
  <n3:airportFromCode>BCN2</n3:airportFromCode>
  <n3:airportToCode>DME</n3:airportToCode>
  <n3:arrivalTime>21:25</n3:arrivalTime>
  <n3:dayShift>0</n3:dayShift>
  <n3:departureTime>14:45</n3:departureTime>
  <n3:ticketNumber xsi:nil="true"/>
</n3:timetable>
</n3:FlightSegment>
</n3:flightSegments>
<n3:grantDate xsi:nil="true"/>
<n3:price>250.0 EUR</n3:price>
<n3:resourceCount>8</n3:resourceCount>
<n3:resourceStatus>A</n3:resourceStatus>
<n3:setName>Москва-Барселона-Москва, Трансаэро</n3:setName>
<n3:tariffCode>A1</n3:tariffCode>
<n3:tariffType>A</n3:tariffType>
<n3:tariffID>24989349483</n3:tariffID>
<n3:templateID>216</n3:templateID>
<n3:tourProductID>577932805</n3:tourProductID>
</n3:Flight>
</n2:flights>
<n4:responseLog xmlns:n4="java:com.nt.slib.ws.wssearch.response">
  <n4:comment>num rows=1</n4:comment>
  <n4:error/>
  <n4:timeForResponse>172</n4:timeForResponse>
</n4:responseLog>
</n1:result>
</m:searchFlightResponse>
</env:Body>
</env:Envelope>
```

Comments:

- *tariffID*: identifier of the Tourproduct price
- *templateID*: identifier of the Tourproduct template; needed to see the Tourproduct items
- *tourProductID*: identifier of the Tourproduct
- *resourceCount*: available allotment
- *resourceStatus* (see *getTourproductTarif*): staus of availability, where:
 - A – available
 - FEW – less than 2 items (rooms, seats) available
 - N/A or null – not available
 - REQ – on request
 - EXPIRED – tariff expired, impossible to book

SearchFlightSpecial

The flights have been created as One ways.

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***/web services"
xmlns:java="java:com.nt.slib.ws.wssearch" xmlns:java1="java:com.nt.slib.ws.wssearch.request">
  <soapenv:Header/>
  <soapenv:Body>
    <web:searchFlightSpecial>
      <web:wsLoginPassword>
        <java:wslogin>***</java:wslogin>
        <java:wspassword>***</java:wspassword>
      </web:wsLoginPassword>
      <web:flightRequest>
        <java1:ageTypeID>101</java1:ageTypeID>
        <java1:airCompanyCode>?</java1:airCompanyCode>
        <java1:airCraftModelID>?</java1:airCraftModelID>
        <java1:airDateFromDestination>01.12.2014</java1:airDateFromDestination>
        <java1:airDateFromOrigin>01.12.2014</java1:airDateFromOrigin>
        <java1:airType>?</java1:airType>
        <java1:classofSeatsGroupTypeID>101</java1:classofSeatsGroupTypeID>
        <java1:classofSeatsTypeID>2</java1:classofSeatsTypeID>
        <java1:destinationAirPortID>?</java1:destinationAirPortID>
        <java1:destinationCityID>822</java1:destinationCityID>
        <java1:flightNumber>525</java1:flightNumber>
        <java1:originAirPortID>?</java1:originAirPortID>
        <java1:originCityID>541</java1:originCityID>
        <java1:priceMax>?</java1:priceMax>
        <java1:priceMin>?</java1:priceMin>
        <java1:rowNumFrom>0</java1:rowNumFrom>
        <java1:rowNumTo>10</java1:rowNumTo>
        <java1:tarifTypeID>?</java1:tarifTypeID>
        <java1:tourProductTemplID>639</java1:tourProductTemplID>
      </web:flightRequest>
    </web:searchFlightSpecial>
  </soapenv:Body>
</soapenv:Envelope>
```

Comments:

Additionally to the *searchFlight* attributes:

- *airCraftModelID*: optional, IDs of airplane types
- *classofSeatsGroupTypeID*: mandatory, class types; business (Id: 102), comfort (Id: 124), first (Id: 103), premium (Id: 144), economy (Id: 101)
- *tourProductTemplID*: ID of Tourproduct Template, mandatory; use 639

Response

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:searchFlightSpecialResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:flights xsi:type="n3:ArrayOfFlight" xmlns:n3="java:com.nt.slib.ws.wssearch.rows"
xmlns:n2="java:com.nt.slib.ws.wssearch.response">
          <n3:Flight>
            <n3:duration>0</n3:duration>
            <n3:flightSegments xsi:type="n3:ArrayOfFlightSegment">
              <n3:FlightSegment>
                <n3:ageType>ADL</n3:ageType>
                <n3:ageTypeID>101</n3:ageTypeID>
                <n3:aircompanyCode>UN</n3:aircompanyCode>
                <n3:classOfSeats>U</n3:classOfSeats>
```

```
<n3:classOfSeatsID>2</n3:classOfSeatsID>
<n3:destinationAirportCode>BKK</n3:destinationAirportCode>
<n3:destinationAirportID>843</n3:destinationAirportID>
<n3:destinationCity>Бангкок</n3:destinationCity>
<n3:flightNumber>525</n3:flightNumber>
<n3:offset>0</n3:offset>
<n3:originAirportCode>DME</n3:originAirportCode>
<n3:originAirportID>840</n3:originAirportID>
<n3:originCity>Москва</n3:originCity>
<n3:timetable>
  <n3:airportFromCode>DME</n3:airportFromCode>
  <n3:airportToCode>BKK</n3:airportToCode>
  <n3:arrivalTime>10:55</n3:arrivalTime>
  <n3:dayShift>1</n3:dayShift>
  <n3:departureTime>22:15</n3:departureTime>
  <n3:ticketNumber xsi:nil="true"/>
</n3:timetable>
</n3:FlightSegment>
</n3:flightSegments>
<n3:grantDate>01.12.2014</n3:grantDate>
<n3:price>300.0 USD</n3:price>
<n3:resourceCount>31</n3:resourceCount>
<n3:resourceStatus>A</n3:resourceStatus>
<n3:setName>Москва-Бангкок, для ***</n3:setName>
<n3:tariffCode>A1</n3:tariffCode>
<n3:tariffType>A</n3:tariffType>
<n3:tariffID>25186320578</n3:tariffID>
<n3:templateID>639</n3:templateID>
<n3:tourProductID>677357854</n3:tourProductID>
</n3:Flight>
<n3:Flight>
  <n3:duration>0</n3:duration>
  <n3:flightSegments xsi:type="n3:ArrayOfFlightSegment">
    <n3:FlightSegment>
      <n3:ageType>ADL</n3:ageType>
      <n3:ageTypeID>101</n3:ageTypeID>
      <n3:aircompanyCode>UN</n3:aircompanyCode>
      <n3:classOfSeats>U</n3:classOfSeats>
      <n3:classOfSeatsID>2</n3:classOfSeatsID>
      <n3:destinationAirportCode>BKK</n3:destinationAirportCode>
      <n3:destinationAirportID>843</n3:destinationAirportID>
      <n3:destinationCity>Бангкок</n3:destinationCity>
      <n3:flightNumber>525</n3:flightNumber>
      <n3:offset>0</n3:offset>
      <n3:originAirportCode>DME</n3:originAirportCode>
      <n3:originAirportID>840</n3:originAirportID>
      <n3:originCity>Москва</n3:originCity>
      <n3:timetable>
        <n3:airportFromCode>DME</n3:airportFromCode>
        <n3:airportToCode>BKK</n3:airportToCode>
        <n3:arrivalTime>10:55</n3:arrivalTime>
        <n3:dayShift>1</n3:dayShift>
        <n3:departureTime>22:15</n3:departureTime>
        <n3:ticketNumber xsi:nil="true"/>
      </n3:timetable>
    </n3:FlightSegment>
  </n3:flightSegments>
  <n3:grantDate>01.12.2014</n3:grantDate>
  <n3:price>335.0 USD</n3:price>
  <n3:resourceCount>31</n3:resourceCount>
  <n3:resourceStatus>A</n3:resourceStatus>
  <n3:setName>Москва-Бангкок для NAIN</n3:setName>
  <n3:tariffCode>A1</n3:tariffCode>
  <n3:tariffType>A</n3:tariffType>
  <n3:tariffID>25414496282</n3:tariffID>
  <n3:templateID>639</n3:templateID>
  <n3:tourProductID>893804802</n3:tourProductID>
</n3:Flight>
</n2:flights>
<n4:responseLog xmlns:n4="java.com.nt.slib.ws.wssearch.response">
  <n4:comment>num rows=2</n4:comment>
  <n4:error/>
  <n4:timeForResponse>85</n4:timeForResponse>
```

```
</n4:responseLog>
</n1:result>
</m:searchFlightSpecialResponse>
</env:Body>
</env:Envelope>
```

getTourProductTariff

To check the availability for every item included in the Tour product (what is actual for roundtrip flights), you will need to make the *getTourProductTariff* request before the booking an order.

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***web services"
xmlns:java="java:com.nt.slib.ws.wssearch" xmlns:java1="java:com.nt.slib.ws.wssearch.request">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getTourProductTariff>
      <web:wsLoginPassword>
        <java:wslogin>***</java:wslogin>
        <java:wspassword>***</java:wspassword>
      </web:wsLoginPassword>
      <web:tourProductTariffRequest>
        <java1:tourProductTariffId>24989349483</java1:tourProductTariffId>
      </web:tourProductTariffRequest>
    </web:getTourProductTariff>
  </soapenv:Body>
</soapenv:Envelope>
```

Comments:

- *tourProductTariffId*= *tariffID* from *searchFlight* Response.

Response

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:getTourProductTariffResponse xmlns:m="http://***web services">
      <n1:result xmlns:n1="http://***web services">
        <n2:responseLog xmlns:n2="java:com.nt.slib.ws.wssearch.response">
          <n2:comment/>
          <n2:error/>
          <n2:timeForResponse>0</n2:timeForResponse>
        </n2:responseLog>
        <n3:tourProductTariffWS xmlns:n3="java:com.nt.slib.ws.wssearch.response">
          <n4:addons xsi:nil="true" xmlns:n4="java:com.nt.slib.ws.wssearch.rows"/>
          <n5:cost xmlns:n5="java:com.nt.slib.ws.wssearch.rows">250.0 EUR</n5:cost>
          <n6:duration xmlns:n6="java:com.nt.slib.ws.wssearch.rows">7</n6:duration>
          <n7:grantDate xmlns:n7="java:com.nt.slib.ws.wssearch.rows">2014-11-22T00:00:00+04:00</n7:grantDate>
          <n8:items xsi:type="n8:ArrayOfTourProductTariffItem" xmlns:n8="java:com.nt.slib.ws.wssearch.rows">
            <n8:TourProductTariffItem>
              <n8:commodityDescription>UN 331, Москва (DME) -> Барселона (BCN2), U, ADL</n8:commodityDescription>
              <n8:commodityType>7</n8:commodityType>
              <n8:cruise xsi:nil="true"/>
              <n8:duration>0</n8:duration>
              <n8:flight xsi:type="n8:FlightCommodity">
                <n8:commodityName>UN 331, Москва -> Барселона, U, Взрослый</n8:commodityName>
                <n8:duration>0</n8:duration>
                <n8:grantDateOffset>0</n8:grantDateOffset>
                <n8:id>226723</n8:id>
                <n8:resourceCount>8</n8:resourceCount>
                <n8:resourceStatus>A</n8:resourceStatus>
                <n8:templateItemId>0</n8:templateItemId>
                <n8:accmdPerson>ADL</n8:accmdPerson>
                <n8:accmdPersonAgeFrom>0</n8:accmdPersonAgeFrom>
                <n8:accmdPersonAgeTo>0</n8:accmdPersonAgeTo>
              </n8:flight>
            </n8:TourProductTariffItem>
          </n8:items>
        </n3:tourProductTariffWS>
      </n1:result>
    </m:getTourProductTariffResponse>
  </env:Body>
</env:Envelope>
```

```
<n8:accmdPersonId>101</n8:accmdPersonId>
<n8:accmdPersonName>Взрослый</n8:accmdPersonName>
<n8:airClassAbbr>U</n8:airClassAbbr>
<n8:airClassId>2</n8:airClassId>
<n8:airClassName>эконом / U</n8:airClassName>
<n8:airTripNumber>UN 331</n8:airTripNumber>
<n8:airTripNumberOnly>331</n8:airTripNumberOnly>
<n8:airTripType>0</n8:airTripType>
<n8:aircraftModelAbbr>777-200UNT</n8:aircraftModelAbbr>
<n8:airlineCode>UN</n8:airlineCode>
<n8:airlineShortName>Transaero</n8:airlineShortName>
<n8:airlineShortNameRus>Трансаэро</n8:airlineShortNameRus>
<n8:airportFromCode xsi:nil="true"/>
<n8:airportFromId>0</n8:airportFromId>
<n8:airportFromName xsi:nil="true"/>
<n8:airportToCode xsi:nil="true"/>
<n8:airportToId>0</n8:airportToId>
<n8:airportToName xsi:nil="true"/>
<n8:arrivalTime xsi:nil="true"/>
<n8:cityFromName>Москва</n8:cityFromName>
<n8:cityFromNameEng xsi:nil="true"/>
<n8:cityToName>Барселона</n8:cityToName>
<n8:cityToNameEng xsi:nil="true"/>
<n8:dayShift>0</n8:dayShift>
<n8:departureDate xsi:nil="true"/>
<n8:departureTime xsi:nil="true"/>
<n8:returnFlightNumber>UN 332</n8:returnFlightNumber>
<n8:returnFlightNumberOnly>332</n8:returnFlightNumberOnly>
<n8:sysActive>1</n8:sysActive>
</n8:flight>
<n8:flightVariants xsi:nil="true"/>
<n8:grantDateOffset>0</n8:grantDateOffset>
<n8:groupExcursion xsi:nil="true"/>
<n8:groupTransfer xsi:nil="true"/>
<n8:individualTransfer xsi:nil="true"/>
<n8:medicalInsurance xsi:nil="true"/>
<n8:miscCommodity xsi:nil="true"/>
<n8:rentaCar xsi:nil="true"/>
<n8:residence xsi:nil="true"/>
<n8:residenceVariants xsi:nil="true"/>
<n8:sectionName>Авианепелет, туда, 22.11</n8:sectionName>
<n8:sightSeeing xsi:nil="true"/>
<n8:variantable>false</n8:variantable>
<n8:visaCommodity xsi:nil="true"/>
</n8:TourProductTariffItem>
<n8:TourProductTariffItem>
<n8:commodityDescription>UN 332, Барселона (BCN2) -> Москва (DME), U, ADL</n8:commodityDescription>
<n8:commodityType>7</n8:commodityType>
<n8:cruise xsi:nil="true"/>
<n8:duration>0</n8:duration>
<n8:flight xsi:type="n8:FlightCommodity">
<n8:commodityName>UN 332, Барселона -> Москва, U, Взрослый</n8:commodityName>
<n8:duration>0</n8:duration>
<n8:grantDateOffset>7</n8:grantDateOffset>
<n8:id>226727</n8:id>
<n8:resourceCount>8</n8:resourceCount>
<n8:resourceStatus>A</n8:resourceStatus>
<n8:templateItemId>0</n8:templateItemId>
<n8:accmdPerson>ADL</n8:accmdPerson>
<n8:accmdPersonAgeFrom>0</n8:accmdPersonAgeFrom>
<n8:accmdPersonAgeTo>0</n8:accmdPersonAgeTo>
<n8:accmdPersonId>101</n8:accmdPersonId>
<n8:accmdPersonName>Взрослый</n8:accmdPersonName>
<n8:airClassAbbr>U</n8:airClassAbbr>
<n8:airClassId>2</n8:airClassId>
<n8:airClassName>эконом / U</n8:airClassName>
<n8:airTripNumber>UN 332</n8:airTripNumber>
<n8:airTripNumberOnly>332</n8:airTripNumberOnly>
<n8:airTripType>0</n8:airTripType>
<n8:aircraftModelAbbr>777-200UNT</n8:aircraftModelAbbr>
<n8:airlineCode>UN</n8:airlineCode>
<n8:airlineShortName>Transaero</n8:airlineShortName>
<n8:airlineShortNameRus>Трансаэро</n8:airlineShortNameRus>
```

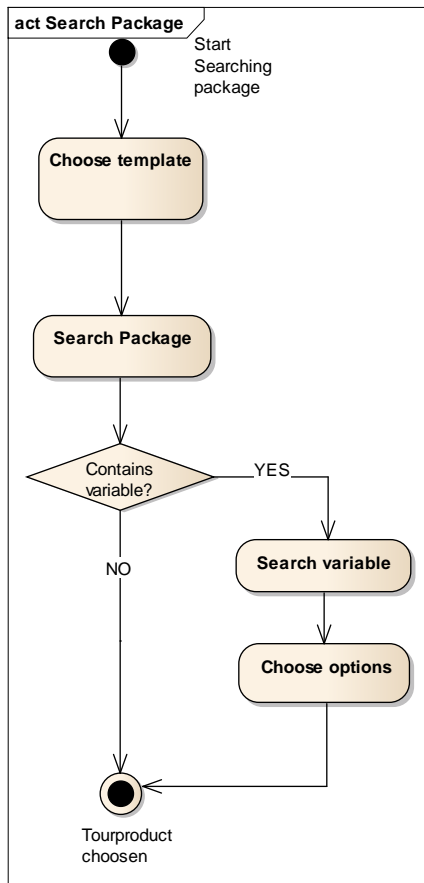
```
<n8:airportFromCode xsi:nil="true"/>
<n8:airportFromId>0</n8:airportFromId>
<n8:airportFromName xsi:nil="true"/>
<n8:airportToCode xsi:nil="true"/>
<n8:airportToId>0</n8:airportToId>
<n8:airportToName xsi:nil="true"/>
<n8:arrivalTime xsi:nil="true"/>
<n8:cityFromName>Барселона</n8:cityFromName>
<n8:cityFromNameEng xsi:nil="true"/>
<n8:cityToName>Москва</n8:cityToName>
<n8:cityToNameEng xsi:nil="true"/>
<n8:dayShift>0</n8:dayShift>
<n8:departureDate xsi:nil="true"/>
<n8:departureTime xsi:nil="true"/>
<n8:returnFlightNumber>UN 331</n8:returnFlightNumber>
<n8:returnFlightNumberOnly>331</n8:returnFlightNumberOnly>
<n8:sysActive>1</n8:sysActive>
</n8:flight>
<n8:flightVariants xsi:nil="true"/>
<n8:grantDateOffset>7</n8:grantDateOffset>
<n8:groupExcursion xsi:nil="true"/>
<n8:groupTransfer xsi:nil="true"/>
<n8:individualTransfer xsi:nil="true"/>
<n8:medicalInsurance xsi:nil="true"/>
<n8:miscCommodity xsi:nil="true"/>
<n8:rentaCar xsi:nil="true"/>
<n8:residence xsi:nil="true"/>
<n8:residenceVariants xsi:nil="true"/>
<n8:sectionName>Авианепелет, обратно, 29.11</n8:sectionName>
<n8:sightSeeing xsi:nil="true"/>
<n8:variantable>false</n8:variantable>
<n8:visaCommodity xsi:nil="true"/>
</n8:TourProductTariffItem>
</n8:items>
<n9:notes xsi:nil="true" xmlns:n9="java:com.nt.slib.ws.wssearch.rows"/>
<n10:tariffBookable xmlns:n10="java:com.nt.slib.ws.wssearch.rows">true</n10:tariffBookable>
<n11:tariffDescription xmlns:n11="java:com.nt.slib.ws.wssearch.rows">A1 (Базовый)</n11:tariffDescription>
<n12:templateId xmlns:n12="java:com.nt.slib.ws.wssearch.rows">216</n12:templateId>
<n13:tourProductName xmlns:n13="java:com.nt.slib.ws.wssearch.rows">Авиабилеты туда и обратно Москва-Барселона-Москва,
Трансаэро 7 н.</n13:tourProductName>
</n3:tourProductTariffWS>
</n1:result>
</m:getTourProductTariffResponse>
</env:Body>
</env:Envelope>
```


Standard package availability request

With these requests, you can search packages with Accommodation as the Main services: both the “Hotel only” (hotels + insurance) packages and the “normal” packages containing flights, transfers and other services.

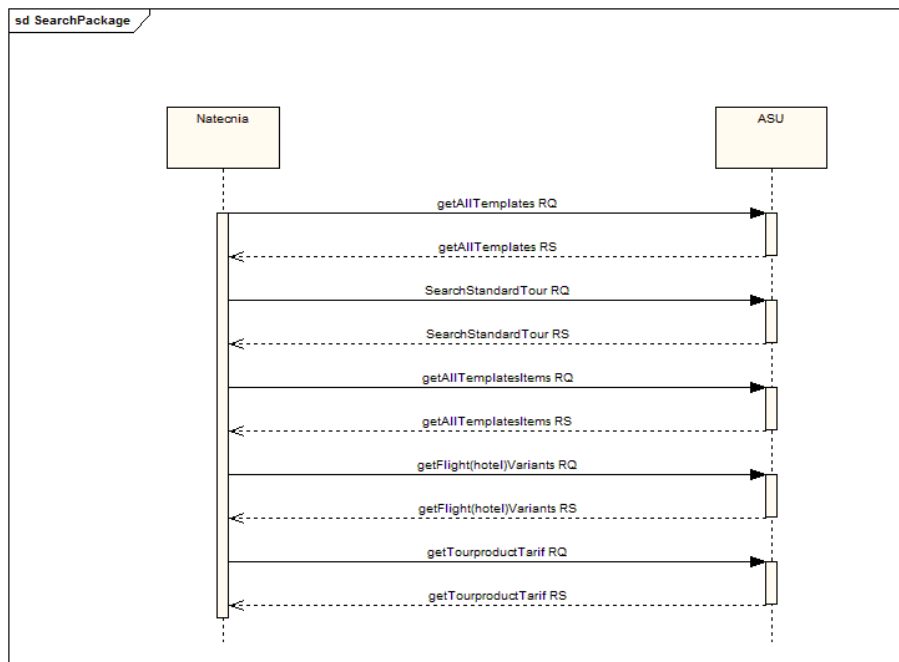
Hotels only and normal packages can be distinguished by *TemplateID* or *withFlightOnly* attributes.

The following Activity diagram describes the workflow:



SearchStandardTour (hotels and packages)

The workflow for normal packages (with flights etc) is more complicate, see the Sequence diagram:



getAllTemplates

Is using to get the template manes and IDs.

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***/web services">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getAllTemplates/>
  </soapenv:Body>
</soapenv:Envelope>
```

Response (fragment)

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:getAllTemplatesResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:errorMessage xsi:nil="true" xmlns:n2="java:com.nt.slib.ws.wssearch.response"/>
        <n3:templates xsi:type="n4:ArrayOfTemplateWS" xmlns:n3="java:com.nt.slib.ws.wssearch.response"
xmlns:n4="java:com.nt.slib.ws.geo">
          <n4:TemplateWS>
            <n4:id>2970</n4:id>
            <n4:name>Корпоративный туризм Тип: "Конференция/Инсентив-7"</n4:name>
          </n4:TemplateWS>
          <n4:TemplateWS>
            <n4:id>3612</n4:id>
            <n4:name>Дополнительная услуга экскурсионный пакет (5 экс.)</n4:name>
          </n4:TemplateWS>
        </n3:templates>
      </n1:result>
    </m:getAllTemplatesResponse>
  </env:Body>
</env:Envelope>
```

```
<n4:TemplateWS>
  <n4:id>4672</n4:id>
  <n4:name>Программа: "СПА-курорты", Тип: "Стандарт"</n4:name>
</n4:TemplateWS>
...
```

searchStandardTour

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***.web services"
xmlns:java="java.com.nt.slib.ws.wssearch" xmlns:java1="java.com.nt.slib.ws.wssearch.request">
  <soapenv:Header/>
  <soapenv:Body>
    <web:searchStandardTour>
      <web:wsLoginPassword>
        <java:wslogin>***</java:wslogin>
        <java:wspassword>***</java:wspassword>
      </web:wsLoginPassword>
      <web:searchStandardTours2Request>
        <java1:boardTypeGroups>?</java1:boardTypeGroups>
        <java1:boardTypes>?</java1:boardTypes>
        <java1:childAges>?</java1:childAges>
        <java1:cityIDs>?</java1:cityIDs>
        <java1:countryID>5</java1:countryID>
        <java1:departureCityId>541</java1:departureCityId>
        <java1:durationMax>7</java1:durationMax>
        <java1:durationMin>7</java1:durationMin>
        <java1:endDate>13.12.2014</java1:endDate>
        <java1:hotelCategoryGroupIDs>?</java1:hotelCategoryGroupIDs>
        <java1:hotelCategoryIDs>?</java1:hotelCategoryIDs>
        <java1:hotelIDs>?</java1:hotelIDs>
        <java1:numberOfAdults>2</java1:numberOfAdults>
        <java1:numberOfChilds>0</java1:numberOfChilds>
        <java1:priceMax>?</java1:priceMax>
        <java1:priceMin>?</java1:priceMin>
        <java1:regionID>?</java1:regionID>
        <java1:rowNumFrom>0</java1:rowNumFrom>
        <java1:rowNumTo>2</java1:rowNumTo>
        <java1:startDate>06.12.2014</java1:startDate>
        <java1:tariffTypeID>?</java1:tariffTypeID>
        <java1:templateGroupID>?</java1:templateGroupID>
        <java1:templateID>1530</java1:templateID>
        <java1:withFlightOnly>?</java1:withFlightOnly>
      </web:searchStandardTours2Request>
    </web:searchStandardTour>
  </soapenv:Body>
</soapenv:Envelope>
```

Response

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:searchStandardTourResponse xmlns:m="http://***.web services">
      <n1:result xmlns:n1="http://***.web services">
        <n2:errorMessage xsi:nil="true" xmlns:n2="java.com.nt.slib.ws.wssearch.response"/>
        <n3:infoMessage xmlns:n3="java.com.nt.slib.ws.wssearch.response">response time=579</n3:infoMessage>
        <n4:lastIndex xmlns:n4="java.com.nt.slib.ws.wssearch.response">5653</n4:lastIndex>
        <n5:rejectedCount xmlns:n5="java.com.nt.slib.ws.wssearch.response">33</n5:rejectedCount>
        <n6:standardTours xsi:type="n7:ArrayOfStandardTour" xmlns:n6="java.com.nt.slib.ws.wssearch.response"
xmlns:n7="java.com.nt.slib.ws.wssearch.rows">
          <n7:StandardTour>
            <n7:airCityFrom>Москва</n7:airCityFrom>
            <n7:airCityFromID>541</n7:airCityFromID>
            <n7:boardTypeID>121</n7:boardTypeID>
```

```
<n7:cost>843.0 EUR</n7:cost>
<n7:duration>7</n7:duration>
<n7:hasRequiredAddon>false</n7:hasRequiredAddon>
<n7:hotel>Los Peces 3 ключа</n7:hotel>
<n7:hotelCity>Испания (Salou)</n7:hotelCity>
<n7:hotelCityID>123</n7:hotelCityID>
<n7:hotelID>24214</n7:hotelID>
<n7:hotelRegionID>970</n7:hotelRegionID>
<n7:resourceStatus>A</n7:resourceStatus>
<n7:roomType>apartments 4 pax</n7:roomType>
<n7:roomTypeID>316</n7:roomTypeID>
<n7:setId>146758</n7:setId>
<n7:setName>Испания (Коста Даурада), Трансаэро</n7:setName>
<n7:startDate>06.12.2014</n7:startDate>
<n7:tarifType>A1</n7:tarifType>
<n7:tarifTypeID>101</n7:tarifTypeID>
<n7:tariffID>24771548621</n7:tariffID>
<n7:templGr xsi:nil="true"/>
<n7:templateID>1530</n7:templateID>
<n7:tourProductID>101539045</n7:tourProductID>
<n7:typeAccommodation>DBL</n7:typeAccommodation>
<n7:typeAccommodationID>102</n7:typeAccommodationID>
</n7:StandardTour>
<n7:StandardTour>
  <n7:airCityFrom>Москва</n7:airCityFrom>
  <n7:airCityFromID>541</n7:airCityFromID>
  <n7:boardTypeID>121</n7:boardTypeID>
  <n7:cost>843.0 EUR</n7:cost>
  <n7:duration>7</n7:duration>
  <n7:hasRequiredAddon>false</n7:hasRequiredAddon>
  <n7:hotel>Los Peces 3 ключа</n7:hotel>
  <n7:hotelCity>Испания (Salou)</n7:hotelCity>
  <n7:hotelCityID>123</n7:hotelCityID>
  <n7:hotelID>24214</n7:hotelID>
  <n7:hotelRegionID>970</n7:hotelRegionID>
  <n7:resourceStatus>A</n7:resourceStatus>
  <n7:roomType>apartments 4 pax</n7:roomType>
  <n7:roomTypeID>316</n7:roomTypeID>
  <n7:setId>146758</n7:setId>
  <n7:setName>Испания (Коста Даурада), Трансаэро</n7:setName>
  <n7:startDate>10.12.2014</n7:startDate>
  <n7:tarifType>A1</n7:tarifType>
  <n7:tarifTypeID>101</n7:tarifTypeID>
  <n7:tariffID>24771571586</n7:tariffID>
  <n7:templGr xsi:nil="true"/>
  <n7:templateID>1530</n7:templateID>
  <n7:tourProductID>101539045</n7:tourProductID>
  <n7:typeAccommodation>DBL</n7:typeAccommodation>
  <n7:typeAccommodationID>102</n7:typeAccommodationID>
</n7:StandardTour>
</n6:standardTours>
<n8:totalMax xmlns:n8="java.com.nt.slib.ws.wssearch.response">5653</n8:totalMax>
</n1:result>
</m:searchStandardTourResponse>
</env:Body>
</env:Envelope>
```

Comments:

- *boardTypeGroups*: optional
- *boardTypes*: optional, see *getAllBoardTypes* request
- *countryID*: mandatory, can be received by using *getAllCountries* request
- *departureCityId*: mandatory, can be received by using *getAllCities* request
- *durationMax*: duration, to... (mandatory)
- *durationMin*: duration, from... (mandatory)
- *endDate*: last date of package start dates range (optional)
- *hotelCategoryGroupIDs*: optional
- *hotelCategoryIDs*: optional, "3*"="3", "3*sup"="4", "4*"="5", "5*"="7"
- *hotelIDs*: optional (no request available)
- *cityIDs*: city of hotel, optional, see *getAllCities* request
- *numberOfAdults*: mandatory
- *numberOfChildren*: mandatory
- *childAges*: mandatory if children are available
- *priceMax*: optional
- *priceMin*: optional
- *regionID*: optional, see *getAllRegions* request
- *rowNumFrom*: mandatory, first number of rows displayed in search results; counted from 0
- *rowNumTo*: mandatory, last number of rows displayed in search results; counted from 1
- *startDate*: first date of package start dates range (mandatory)
- *tariffTypeID*: optional, use *getTariffTypes* request
- *templateGroupID*: Id of Template group, optional
- *withFlightOnly*: hotels only or normal packages (with flights), optional; true or false
- *templateID*: mandatory, use *getAllTemplates* request; 214 for standard tours, 219 for beach hotels only, 218 for city breaks (packages), 221 for city hotels only, 237 for standard ski tours, 238 for ski hotels only
- *hasRequiredAddon*: true/false: shows if there are any mandatory extras available, to this Tourproduct

getAllTemplateItems

To get the list of the Tourproduct items, and know, which services are variant (have options which do not change the total price of the package – e.g. optional charter flights), we need to make a *getAllTemplateItems* request, from *BookingWS*.

Request

```
<soapenv:Body>
  <web:getAllTemplateItems>
    <web:wsLoginPassword>
      <java:wslogin>***</java:wslogin>
      <java:wspassword>***</java:wspassword>
    </web:wsLoginPassword>
    <web:templateItemRequest>
      <java1:templateID>1530</java1:templateID>
    </web:templateItemRequest>
  </web:getAllTemplateItems>
</soapenv:Body>
```

Response

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:getAllTemplateItemsResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:responseLog xmlns:n2="java:com.nt.slib.ws.wssearch.response">
          <n2:comment/>
          <n2:error/>
          <n2:timeForResponse>0</n2:timeForResponse>
        </n2:responseLog>
        <n3:templateItems xsi:type="n4:ArrayOfTemplateItem" xmlns:n3="java:com.nt.slib.ws.wssearch.response"
          xmlns:n4="java:com.nt.slib.ws.wssearch.rows">
          <n4:TemplateItem>
            <n4:commodityTypeID>1</n4:commodityTypeID>
```

```
<n4:commodityTypeName>Трансфер групповой, отель-ап</n4:commodityTypeName>
<n4:dependOnItemId>0</n4:dependOnItemId>
<n4:id>1970</n4:id>
<n4:multiple>>false</n4:multiple>
</n4:TemplateItem>
<n4:TemplateItem>
  <n4:commodityTypeId>1</n4:commodityTypeId>
  <n4:commodityTypeName>Трансфер групповой, ап-отель</n4:commodityTypeName>
  <n4:dependOnItemId>0</n4:dependOnItemId>
  <n4:id>1972</n4:id>
  <n4:multiple>>false</n4:multiple>
</n4:TemplateItem>
<n4:TemplateItem>
  <n4:commodityTypeId>2</n4:commodityTypeId>
  <n4:commodityTypeName>Мед. страховка,</n4:commodityTypeName>
  <n4:dependOnItemId>0</n4:dependOnItemId>
  <n4:id>1973</n4:id>
  <n4:multiple>>false</n4:multiple>
</n4:TemplateItem>
<n4:TemplateItem>
  <n4:commodityTypeId>3</n4:commodityTypeId>
  <n4:commodityTypeName>Прочая услуга, услуги гида</n4:commodityTypeName>
  <n4:dependOnItemId>0</n4:dependOnItemId>
  <n4:id>1971</n4:id>
  <n4:multiple>>false</n4:multiple>
</n4:TemplateItem>
<n4:TemplateItem>
  <n4:commodityTypeId>5</n4:commodityTypeId>
  <n4:commodityTypeName>Проживание,</n4:commodityTypeName>
  <n4:dependOnItemId>0</n4:dependOnItemId>
  <n4:id>1974</n4:id>
  <n4:multiple>>false</n4:multiple>
</n4:TemplateItem>
<n4:TemplateItem>
  <n4:commodityTypeId>7</n4:commodityTypeId>
  <n4:commodityTypeName>Авианперелет, туда</n4:commodityTypeName>
  <n4:dependOnItemId>1969</n4:dependOnItemId>
  <n4:id>1975</n4:id>
  <n4:multiple>>true</n4:multiple>
</n4:TemplateItem>
<n4:TemplateItem>
  <n4:commodityTypeId>7</n4:commodityTypeId>
  <n4:commodityTypeName>Авианперелет, обратно</n4:commodityTypeName>
  <n4:dependOnItemId>0</n4:dependOnItemId>
  <n4:id>1969</n4:id>
  <n4:multiple>>true</n4:multiple>
</n4:TemplateItem>
</n3:templateItems>
</n1:result>
</m:getAllTemplateItemsResponse>
</env:Body>
</env:Envelope>
```

Comments:

In the response, you are getting an array of the Tourproduct items/
Each Item has the following attributes:

- *commodityTypeId*: service type ID (from 1 to 13, e.g. Accommodation=5, Flight=7 etc.)
- *commodityTypeName*: "commodity type name" + "," + "Template item name"
- *id*: Tourproduct Item ID
- *multiple*: variability indicator, true/false

Getting Options

To get the list of the options for each variable item (*multiple*=true), depending on the service type, we need to fulfill the certain request for each *templateItemId* (for Accommodation, *getResidenceVariants*; for Flight, *getFlightVariants* etc).

getFlightVariants

Request

```
<web:getFlightVariants>
  <web:wsLoginPassword>
    <java:wslogin>***</java:wslogin>
    <java:wspassword>***</java:wspassword>
  </web:wsLoginPassword>
</web:wsLoginPassword>
  <web:flightCommodityRequest>
    <java1:templateItemId>1975</java1:templateItemId>
    <java1:tourProductTariffId>21545506185</java1:tourProductTariffId>
  </web:flightCommodityRequest>
</web:getFlightVariants>
```

Comments:

- *tourProductTariffId* = tariffID from the *searchStandardTour* response

Response (fragment)

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Body>
    <m:getFlightVariantsResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:commodities xsi:type="n3:ArrayOfFlightCommodity" xmlns:n3="java:com.nt.slib.ws.wssearch.rows"
xmlns:n2="java:com.nt.slib.ws.wssearch.response">
          <n3:FlightCommodity xsi:type="n3:FlightCommodity">
            <n3:commodityName>S7 885, Москва -> Плайя де Лас Америкас, Y, Взрослый</n3:commodityName>
            <n3:duration>0</n3:duration>
            <n3:grantDateOffset>0</n3:grantDateOffset>
            <n3:id>130085</n3:id>
            <n3:resourceCount>0</n3:resourceCount>
            <n3:resourceStatus xsi:nil="true"/>
            <n3:templateItemId>0</n3:templateItemId>
            <n3:accmdPerson>ADL</n3:accmdPerson>
            <n3:accmdPersonAgeFrom>0</n3:accmdPersonAgeFrom>
            <n3:accmdPersonAgeTo>0</n3:accmdPersonAgeTo>
            <n3:accmdPersonId>101</n3:accmdPersonId>
            <n3:accmdPersonName>Взрослый</n3:accmdPersonName>
            <n3:airClassAbbr>Y</n3:airClassAbbr>
            <n3:airClassId>1</n3:airClassId>
            <n3:airClassName>эконом класс / Y</n3:airClassName>
            <n3:airTripNumber>S7 885</n3:airTripNumber>
            <n3:airTripNumberOnly>885</n3:airTripNumberOnly>
            <n3:airTripType>1</n3:airTripType>
            <n3:aircraftModelAbbr>B-767-200</n3:aircraftModelAbbr>
            <n3:airlineCode>S7</n3:airlineCode>
            <n3:airlineShortName>Sibir</n3:airlineShortName>
            <n3:airlineShortNameRus>Сибирь</n3:airlineShortNameRus>
            <n3:airportFromCode>DME</n3:airportFromCode>
            <n3:airportFromId>840</n3:airportFromId>
            <n3:airportFromName>Domodedovo</n3:airportFromName>
            <n3:airportToCode>TFS</n3:airportToCode>
            <n3:airportToId>846</n3:airportToId>
            <n3:airportToName>Tenerife</n3:airportToName>
            <n3:arrivalTime xsi:nil="true"/>
            <n3:cityFromName>Москва</n3:cityFromName>
            <n3:cityFromNameEng xsi:nil="true"/>
            <n3:cityToName>Плайя де Лас Америкас</n3:cityToName>
            <n3:cityToNameEng xsi:nil="true"/>
            <n3:dayShift>0</n3:dayShift>
            <n3:departureDate xsi:nil="true"/>
            <n3:departureTime xsi:nil="true"/>
            <n3:returnFlightNumber>S7 886</n3:returnFlightNumber>
            <n3:returnFlightNumberOnly>886</n3:returnFlightNumberOnly>
```

```
<n3:sysActive>1</n3:sysActive>
</n3:FlightCommodity>
<n3:FlightCommodity xsi:type="n3:FlightCommodity">
  <n3:commodityName>NN 771, Москва -> Плайя де Лас Америкас, Y, Взрослый</n3:commodityName>
  <n3:duration>0</n3:duration>
  <n3:grantDateOffset>0</n3:grantDateOffset>
  <n3:id>130105</n3:id>
  <n3:resourceCount>0</n3:resourceCount>
  <n3:resourceStatus xsi:nil="true"/>
  <n3:templateItemId>0</n3:templateItemId>
  <n3:accmdPerson>ADL</n3:accmdPerson>
  <n3:accmdPersonAgeFrom>0</n3:accmdPersonAgeFrom>
  <n3:accmdPersonAgeTo>0</n3:accmdPersonAgeTo>
  <n3:accmdPersonId>101</n3:accmdPersonId>
  <n3:accmdPersonName>Взрослый</n3:accmdPersonName>
  <n3:airClassAbbr>Y</n3:airClassAbbr>
  <n3:airClassId>1</n3:airClassId>
  <n3:airClassName>эконом класс / Y</n3:airClassName>
  <n3:airTripNumber>NN 771</n3:airTripNumber>
  <n3:airTripNumberOnly>771</n3:airTripNumberOnly>
  <n3:airTripType>1</n3:airTripType>
  <n3:aircraftModelAbbr>B-767-200</n3:aircraftModelAbbr>
  <n3:airlineCode>NN</n3:airlineCode>
  <n3:airlineShortName>VIM-AVIA</n3:airlineShortName>
  <n3:airlineShortNameRus>VIM-AVIA</n3:airlineShortNameRus>
  <n3:airportFromCode>DME</n3:airportFromCode>
  <n3:airportFromId>840</n3:airportFromId>
  <n3:airportFromName>Domodedovo</n3:airportFromName>
  <n3:airportToCode>TFS</n3:airportToCode>
  <n3:airportToId>846</n3:airportToId>
  <n3:airportToName>Tenerife</n3:airportToName>
  <n3:arrivalTime xsi:nil="true"/>
  <n3:cityFromName>Москва</n3:cityFromName>
  <n3:cityFromNameEng xsi:nil="true"/>
  <n3:cityToName>Плайя де Лас Америкас</n3:cityToName>
  <n3:cityToNameEng xsi:nil="true"/>
  <n3:dayShift>0</n3:dayShift>
  <n3:departureDate xsi:nil="true"/>
  <n3:departureTime xsi:nil="true"/>
  <n3:returnFlightNumber>NN 772</n3:returnFlightNumber>
  <n3:returnFlightNumberOnly>772</n3:returnFlightNumberOnly>
  <n3:sysActive>1</n3:sysActive>
</n3:FlightCommodity>
```

...

Comments:

- *grantDateOffset*: maximum item's start date shift in the Template (can be 0 or 1 – in case of flights, to make available the same or next day departures)

getTourProductTariff

To check the availability for the items included in the Tourproduct, you will need to make the *getTourProductTariff* request before the booking an order.

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***/web services"
xmlns:java="java.com.nt.slib.ws.wssearch" xmlns:java1="java.com.nt.slib.ws.wssearch.request">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getTourProductTariff>
      <web:wsLoginPassword>
        <java:wslogin>***</java:wslogin>
        <java:wspassword>***</java:wspassword>
      </web:wsLoginPassword>
      <web:tourProductTariffRequest>
        <java1:tourProductTariffId>24771548621</java1:tourProductTariffId>
      </web:tourProductTariffRequest>
    </web:getTourProductTariff>
  </soapenv:Body>
</soapenv:Envelope>
```

Response (fragment)

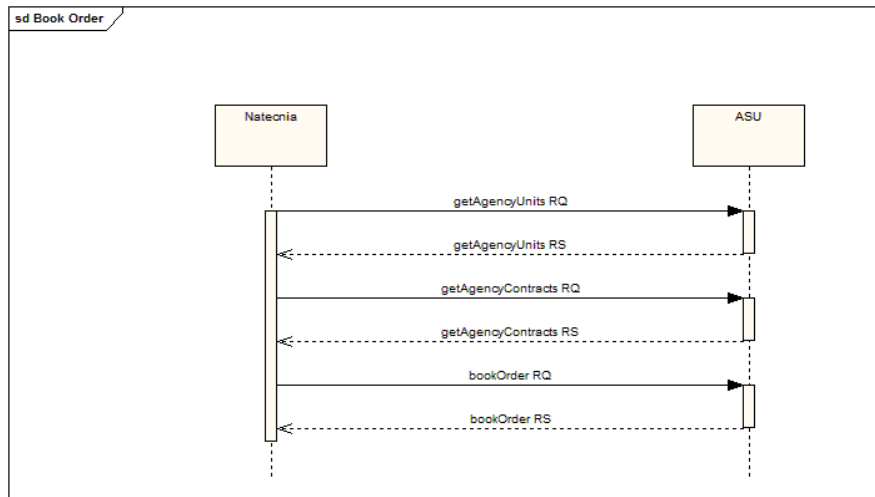
```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:getTourProductTariffResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:responseLog xmlns:n2="java.com.nt.slib.ws.wssearch.response">
          <n2:comment/>
          <n2:error/>
          <n2:timeForResponse>0</n2:timeForResponse>
        </n2:responseLog>
        <n3:tourProductTariffWS xmlns:n3="java.com.nt.slib.ws.wssearch.response">
          <n4:addons xsi:nil="true" xmlns:n4="java.com.nt.slib.ws.wssearch.rows"/>
          <n5:cost xmlns:n5="java.com.nt.slib.ws.wssearch.rows">843.0 EUR</n5:cost>
          <n6:duration xmlns:n6="java.com.nt.slib.ws.wssearch.rows">7</n6:duration>
          <n7:grantDate xmlns:n7="java.com.nt.slib.ws.wssearch.rows">2014-12-06T00:00:00+04:00</n7:grantDate>
          <n8:items xsi:type="n8:ArrayOfTourProductTariffItem" xmlns:n8="java.com.nt.slib.ws.wssearch.rows">
            <n8:TourProductTariffItem>
              <n8:commodityDescription>DBL | AO | apartments 4 pax, Los Peces 3 ключа, Салоу, Испания</n8:commodityDescription>
              <n8:commodityType>5</n8:commodityType>
              <n8:cruise xsi:nil="true"/>
              <n8:duration>7</n8:duration>
              <n8:flight xsi:nil="true"/>
              <n8:flightVariants xsi:nil="true"/>
              <n8:grantDateOffset>0</n8:grantDateOffset>
              <n8:groupExcursion xsi:nil="true"/>
              <n8:groupTransfer xsi:nil="true"/>
              <n8:individualTransfer xsi:nil="true"/>
              <n8:medicalInsurance xsi:nil="true"/>
              <n8:miscCommodity xsi:nil="true"/>
              <n8:rentaCar xsi:nil="true"/>
              <n8:residence xsi:type="n8:ResidenceCommodity">
                <n8:commodityName>DBL | AO | apartments 4 pax, Los Peces 3 ключа, Салоу, Испания</n8:commodityName>
                <n8:duration>7</n8:duration>
                <n8:grantDateOffset>0</n8:grantDateOffset>
                <n8:id>75868</n8:id>
                <n8:resourceCount>10</n8:resourceCount>
                <n8:resourceStatus>A</n8:resourceStatus>
                <n8:templateItemId>0</n8:templateItemId>
              </n8:residence>
            </n8:TourProductTariffItem>
          </n8:items>
        </n3:tourProductTariffWS>
      </n1:result>
    </m:getTourProductTariffResponse>
  </env:Body>
</env:Envelope>
```

```
<n8:accmdTypeAbbr>DBL</n8:accmdTypeAbbr>
<n8:accmdTypeId>102</n8:accmdTypeId>
<n8:boardTypeAbbr>AO</n8:boardTypeAbbr>
<n8:boardTypeId>121</n8:boardTypeId>
<n8:categoryId>321</n8:categoryId>
<n8:categoryName>3 ключа</n8:categoryName>
<n8:chainId>0</n8:chainId>
<n8:chainName xsi:nil="true"/>
<n8:cityId>123</n8:cityId>
<n8:cityNameEng>Salou</n8:cityNameEng>
<n8:cityNameRus>Салю</n8:cityNameRus>
<n8:countryId>5</n8:countryId>
<n8:countryName>Испания</n8:countryName>
<n8:hotelId>24214</n8:hotelId>
<n8:hotelRoomId>26852</n8:hotelRoomId>
<n8:regionId>970</n8:regionId>
<n8:regionNameEng>Costa Daurada</n8:regionNameEng>
<n8:regionNameRus>Коста Даурда</n8:regionNameRus>
<n8:residenceName>Los Peces</n8:residenceName>
<n8:residenceTypeName>отель</n8:residenceTypeName>
<n8:roomTypeAbbr>apartments 4 pax</n8:roomTypeAbbr>
<n8:roomTypeId>316</n8:roomTypeId>
<n8:roomTypeName>apartments 4 pax</n8:roomTypeName>
<n8:sysActive>1</n8:sysActive>
</n8:residence>
<n8:residenceVariants xsi:nil="true"/>
<n8:sectionName>Проживание, с 06.12 по 13.12</n8:sectionName>
<n8:sightSeeing xsi:nil="true"/>
<n8:variantable>false</n8:variantable>
<n8:visaCommodity xsi:nil="true"/>
</n8:TourProductTariffItem>
<n8:TourProductTariffItem>
  <n8:commodityDescription xsi:nil="true"/>
  <n8:commodityType>7</n8:commodityType>
  <n8:cruise xsi:nil="true"/>
  <n8:duration>0</n8:duration>
  <n8:flight xsi:nil="true"/>
  <n8:flightVariants xsi:type="n8:ArrayOfFlightCommodity">
    <n8:FlightCommodity xsi:type="n8:FlightCommodity">
      <n8:commodityName>UN 331, Москва -> Барселона, U, Взрослый</n8:commodityName>
      <n8:duration>0</n8:duration>
      <n8:grantDateOffset>0</n8:grantDateOffset>
      <n8:id>226723</n8:id>
      <n8:resourceCount>68</n8:resourceCount>
      <n8:resourceStatus>A</n8:resourceStatus>
      <n8:templateItemId>0</n8:templateItemId>
      <n8:accmdPerson>ADL</n8:accmdPerson>
      <n8:accmdPersonAgeFrom>0</n8:accmdPersonAgeFrom>
      <n8:accmdPersonAgeTo>0</n8:accmdPersonAgeTo>
      <n8:accmdPersonId>101</n8:accmdPersonId>
      <n8:accmdPersonName>Взрослый</n8:accmdPersonName>
      <n8:airClassAbbr>U</n8:airClassAbbr>
      <n8:airClassId>2</n8:airClassId>
      <n8:airClassName>эконом / U</n8:airClassName>
      <n8:airTripNumber>UN 331</n8:airTripNumber>
      <n8:airTripNumberOnly>331</n8:airTripNumberOnly>
      <n8:airTripType>0</n8:airTripType>
      <n8:aircraftModelAbbr>777-200UNT</n8:aircraftModelAbbr>
      <n8:airlineCode>UN</n8:airlineCode>
      <n8:airlineShortName>Transaero</n8:airlineShortName>
      <n8:airlineShortNameRus>Трансаэро</n8:airlineShortNameRus>
      <n8:airportFromCode xsi:nil="true"/>
      <n8:airportFromId>840</n8:airportFromId>
      <n8:airportFromName xsi:nil="true"/>
      <n8:airportToCode xsi:nil="true"/>
      <n8:airportToId>839</n8:airportToId>
      <n8:airportToName xsi:nil="true"/>
      <n8:arrivalTime>13:25</n8:arrivalTime>
      <n8:cityFromName>Москва</n8:cityFromName>
      <n8:cityFromNameEng xsi:nil="true"/>
      <n8:cityToName>Барселона</n8:cityToName>
      <n8:cityToNameEng xsi:nil="true"/>
      <n8:dayShift>0</n8:dayShift>
```

```
<n8:departureDate>06.12.14</n8:departureDate>
<n8:departureTime>10:45</n8:departureTime>
<n8:returnFlightNumber>UN 332</n8:returnFlightNumber>
<n8:returnFlightNumberOnly>332</n8:returnFlightNumberOnly>
<n8:sysActive>1</n8:sysActive>
</n8:FlightCommodity>
...
```

Book Order workflow

To create an order, you need to make the following operations:



getAgencyUnits

Use this request to get the list of your Agency' Units (point of sales) and choose the Agency Unit ID (*id* in the Response).

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***/web services"
xmlns:java="java:com.nt.slib.ws.wssearch">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getAgencyUnits>
      <web:wsBookingLoginPassword>
        <java:wslogin?></java:wslogin>
        <java:wspassword?></java:wspassword>
      </web:wsBookingLoginPassword>
    </web:getAgencyUnits>
  </soapenv:Body>
</soapenv:Envelope>
```

Response (fragment)

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:getAgencyUnitsResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:responseLog xmlns:n2="java:com.nt.slib.ws.wssearch.booking.protocol.info">
          <n3:comment xmlns:n3="java:com.nt.slib.ws.wssearch.response"/>
          <n4:error xmlns:n4="java:com.nt.slib.ws.wssearch.response"/>
          <n5:timeForResponse xmlns:n5="java:com.nt.slib.ws.wssearch.response">0</n5:timeForResponse>
        </n2:responseLog>
        <n6:units xsi:type="n7:ArrayOfAgencyUnitWS" xmlns:n6="java:com.nt.slib.ws.wssearch.booking.protocol.info"
xmlns:n7="java:com.nt.slib.ws.wssearch.booking.model">
          <n7:AgencyUnitWS>
            <n7:id>89403</n7:id>
            <n7:name>Camapa</n7:name>
          </n7:AgencyUnitWS>
          <n7:AgencyUnitWS>
            <n7:id>28877</n7:id>
            <n7:name>Маяковка</n7:name>
          </n7:AgencyUnitWS>
          ...
        </n6:units>
      </n1:result>
    </m:getAgencyUnitsResponse>
  </env:Body>
</env:Envelope>
```

getAgencyContracts

Use this request to get the list of your Agency' Contracts and choose the Agency Contract ID (id in the Response). *ExpiryDate* has to be later than today.

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***/web services"
xmlns:java="java:com.nt.slib.ws.wssearch">
  <soapenv:Header/>
  <soapenv:Body>
    <web:getAgencyContracts>
      <web:wsBookingLoginPassword>
        <java:wslogin>***</java:wslogin>
        <java:wspassword>***</java:wspassword>
      </web:wsBookingLoginPassword>
    </web:getAgencyContracts>
  </soapenv:Body>
</soapenv:Envelope>
```

Response (fragment)

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <env:Header/>
  <env:Body>
    <m:getAgencyContractsResponse xmlns:m="http://***/web services">
      <n1:result xmlns:n1="http://***/web services">
        <n2:contracts xsi:type="n3:ArrayOfAgencyContractWS" xmlns:n3="java:com.nt.slib.ws.wssearch.booking.model"
xmlns:n2="java:com.nt.slib.ws.wssearch.booking.protocol.info">
          <n3:AgencyContractWS>
            <n3:bankName xsi:nil="true"/>
            <n3:contractBlank>Договор нефирменного агентства</n3:contractBlank>
            <n3:contractNumber>тыры</n3:contractNumber>
            <n3:corAccount xsi:nil="true"/>
            <n3:expireDate>2015-12-31T00:00:00+04:00</n3:expireDate>
            <n3:grantDate>2004-01-01T00:00:00+03:00</n3:grantDate>
            <n3:id>494858</n3:id>
            <n3:inn>7701150631</n3:inn>
          </n3:AgencyContractWS>
        </n2:contracts>
      </n1:result>
    </m:getAgencyContractsResponse>
  </env:Body>
</env:Envelope>
```

```
<n3:kpp>45196928</n3:kpp>
<n3:okpo xsi:nil="true"/>
<n3:setAccount xsi:nil="true"/>
</n3:AgencyContractWS>
...
</n2:contracts>
<n4:responseLog xmlns:n4="java:com.nt.slib.ws.wssearch.booking.protocol.info">
  <n5:comment xmlns:n5="java:com.nt.slib.ws.wssearch.response"/>
  <n6:error xmlns:n6="java:com.nt.slib.ws.wssearch.response"/>
  <n7:timeForResponse xmlns:n7="java:com.nt.slib.ws.wssearch.response">0</n7:timeForResponse>
</n4:responseLog>
</n1:result>
</m:getAgencyContractsResponse>
</env:Body>
</env:Envelope>
```

bookOrder

The petition consists of the following sections:

- *wsBookingLoginPassword*: here, is it to mention the login, password, Agency Unit ID and Contract ID
- *firstOrderSaveRequest*: to input passenger data
- *variantsRequest*: to input options, for the variable services
- *addonsRequest*: to input the additional services

Request

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:web="http://***/web services"
xmlns:java="java:com.nt.slib.ws.wssearch" xmlns:java1="java:com.nt.slib.ws.wssearch.booking.protocol.first"
xmlns:java2="java:com.nt.slib.ws.wssearch.booking.model">
  <soapenv:Header/>
  <soapenv:Body>
    <web:bookOrder>
      <web:wsBookingLoginPassword>
        <ns1:idAgencyUnit>146607</ns1:idAgencyUnit>
        <ns1:idContract>84547223</ns1:idContract>
        < ns1:wslogin>***</java:wslogin>
        < ns1:wspassword>***</java:wspassword>
      </web:wsBookingLoginPassword>
      <ns2:firstOrderSaveRequest>
        <ns4:passengerWSs>
          <ns3:PassengerWS>
            <ns3:address xsi:nil="true"/>
            <ns3:birthPlace xsi:nil="true"/>
            <ns3:birthday>1991-11-30T00:00:00</ns3:birthday>
            <ns3:email xsi:nil="true"/>
            <ns3:firstname>EKATERINA</ns3:firstname>
            <ns3:flyWithId xsi:nil="true"/>
            <ns3:foreignPassPlace xsi:nil="true"/>
            <ns3:foreignPassReleaseDate xsi:nil="true"/>
            <ns3:foreignPassSeries xsi:nil="true"/>
            <ns3:foreignPassportExpiryDate>2020-10-10T00:00:00</ns3:foreignPassportExpiryDate>
            <ns3:foreingPasspNum>723736978</ns3:foreingPasspNum>
            <ns3:lastname>KAZAKOVA</ns3:lastname>
            <ns3:lastnameRU xsi:nil="true"/>
            <ns3:nameRU xsi:nil="true"/>
            <ns3:nationalPasspNumber xsi:nil="true"/>
            <ns3:nationalPasspSeries xsi:nil="true"/>
            <ns3:passengerID>-1</ns3:passengerID>
            <ns3:passengerType>3</ns3:passengerType>
            <ns3:payClientType xsi:nil="true"/>
            <ns3:phone1 xsi:nil="true"/>
            <ns3:phone2 xsi:nil="true"/>
            <ns3:phoneMob xsi:nil="true"/>
            <ns3:putInPassportId xsi:nil="true"/>
          </ns3:PassengerWS>
        </ns4:passengerWSs>
        <ns4:tourProductWS>
```

```
<ns3:id>279773484</ns3:id>
<ns3:idTariff>21545506185</ns3:idTariff>
</ns4:tourProductWS>
</ns2:firstOrderSaveRequest>
<ns2:variantsRequest>
  <ns4:flightVarWSs>
<ns3:VariantsWS>
<ns3:commodityID>220483</ns3:commodityID>
<ns3:passengerID>-1</ns3:passengerID>
<ns3:templateItemID>1975</ns3:templateItemID>
</ns3:VariantsWS>
<ns3:VariantsWS>
<ns3:commodityID>220503</ns3:commodityID>
<ns3:passengerID>-1</ns3:passengerID>
<ns3:templateItemID>1969</ns3:templateItemID>
</ns3:VariantsWS>
  </ns4:flightVarWSs>
  <ns4:groupExcursionVarWSs xsi:nil="true"/>
  <ns4:residenceVarWSs xsi:nil="true"/>
  <ns4:sightseeingVarWSs xsi:nil="true"/>
</ns2:variantsRequest>
<web:addonsRequest>
  <java1:carrierRentAddonWSs>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
      <java2:passengerID>?</java2:passengerID>
      <java2:templateAddonID>?</java2:templateAddonID>
      <java2:tourProductTariffID>?</java2:tourProductTariffID>
    </java2:AddonWS>
  </java1:carrierRentAddonWSs>
  <java1:flightAddonWS>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
      <java2:passengerID>?</java2:passengerID>
      <java2:templateAddonID>?</java2:templateAddonID>
      <java2:tourProductTariffID>?</java2:tourProductTariffID>
    </java2:AddonWS>
  </java1:flightAddonWS>
  <java1:groupExcursionAddonWSs>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
      <java2:passengerID>?</java2:passengerID>
      <java2:templateAddonID>?</java2:templateAddonID>
      <java2:tourProductTariffID>?</java2:tourProductTariffID>
    </java2:AddonWS>
  </java1:groupExcursionAddonWSs>
  <java1:individualTransferAddonWSs>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
      <java2:passengerID>?</java2:passengerID>
      <java2:templateAddonID>?</java2:templateAddonID>
      <java2:tourProductTariffID>?</java2:tourProductTariffID>
    </java2:AddonWS>
  </java1:individualTransferAddonWSs>
  <java1:miscCommodityAddonWSs>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
      <java2:passengerID>?</java2:passengerID>
      <java2:templateAddonID>?</java2:templateAddonID>
      <java2:tourProductTariffID>?</java2:tourProductTariffID>
    </java2:AddonWS>
  </java1:miscCommodityAddonWSs>
  <java1:residenceAddonWSs>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
      <java2:passengerID>?</java2:passengerID>
      <java2:templateAddonID>?</java2:templateAddonID>
      <java2:tourProductTariffID>?</java2:tourProductTariffID>
    </java2:AddonWS>
  </java1:residenceAddonWSs>
  <java1:visaAddonWSs>
    <!--Zero or more repetitions:-->
    <java2:AddonWS>
```

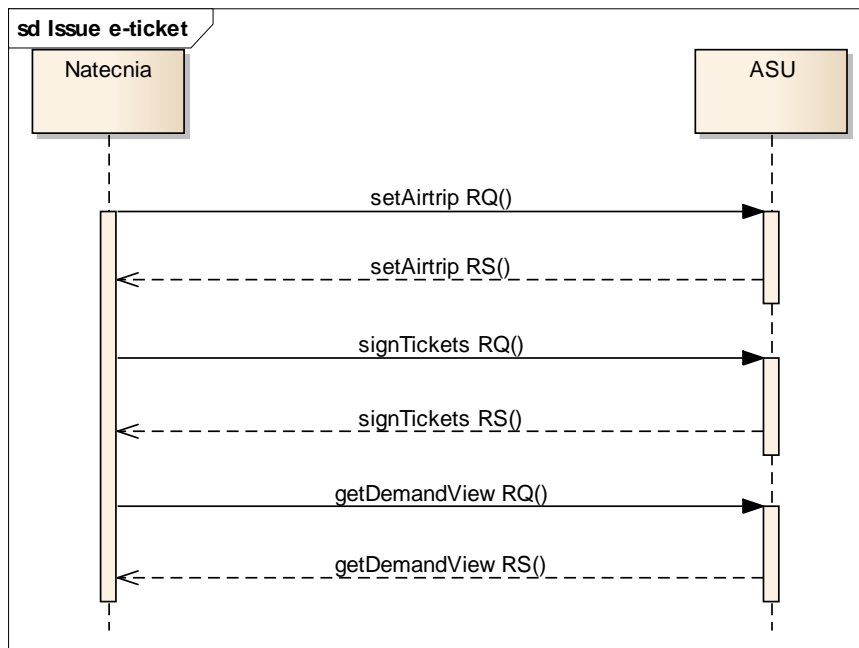
```
<java2:passengerID>?</java2:passengerID>  
<java2:templateAddonID>?</java2:templateAddonID>  
<java2:tourProductTariffID>?</java2:tourProductTariffID>  
</java2:AddonWS>  
</java1:visaAddonWSs>  
</web:addonsRequest>  
</web:bookOrder>  
</soapenv:Body>  
</soapenv:Envelope>
```

Comments:

- In *firstOrderSaveRequest* section, *passengerID* have to be inputted as negative for the first Order saving
- *tourProductWS.idTariff* = *tariffID*
- *tourProductWS.id* = *tourProductID*
- In *variansRequest*, *commodityID* = *id* of the option, from *getFlightVariants* Response; *passengerID* = *passengerID* from *firstOrderSaveRequest* section

E-tickets issue

The sequence diagram for this process is:



setAirtrip

Before issuing e-tickets, it is necessary to set "final" status for Flights, to forbid any changes before the issuing. To make this, set the *isReady* attribute to *true* (to allow changes of the flight again, set *isReady* attribute to *false*).

The operation had been completed successfully if the system returns no errors.

signTickets

The only attribute is *orderId*.

The operation had been completed successfully if the system returns no errors.

getDemandView

To obtain the e-ticket number, it is to make the *getDemandView* request.

Booking modifications

The following kinds of modifications are supported:

- Add a Package to the existing Order with *addTourproduct* request;
- Update Passenger personal data using *updatePassenger* request.

Booking cancellation

You can cancel either one Product (Service) from the Order with *cancelOneTourproduct* request, or the Order itself with *cancelOrder* request.