



Documentatie tehnica API UrgentOnline

<https://urgentcargus.portal.azure-api.net/>

Tehnologie folosita: **REST Web Api Microsoft**

V1.0

Contents

Contents.....	1
1Inregistrare la API StandardUrgentOnlineAPI.....	4

2Exemplu de apel PHP:	11
3Autentificare	11
3.1Logare si obtinere token	11
3.2Verificarea tokenului	12
4Geografie	12
4.1Lista cu tarile din sistem	12
4.2Lista judetelor dintr-o tara data	13
4.3Lista localitatilor dintr-un judet	13
4.4Lista strazilor dintr-o localitate	14
5Punctele de ridicare	14
5.1Lista punctelor de ridicare pe client	14
5.2Lista partenerilor Collect Shop	15
5.3Activarea (asignarea) unui punct de ridicare pe user	16
5.4Listarea punctelor de ridicare active pe user	16
5.5Adaugarea unui punct de ridicare pe un utilizator	17
5.6Modificarea unui punct de ridicare	18
6Tarif	19
6.1Lista tarifelor din contract	19
6.2Calcul tarif pentru o expediere	20
7Administrarea notelor de transport	21
7.1Generarea notelor de transport	21
7.2Generarea notelor de transport in sistem Collect Shop	22
7.3Stergerea unei note de transport	23
7.4Listarea notelor dintr-o anumita perioada	24
7.5Listare informatii note transport pentru o anumita comanda	25
7.6Tiparire note de transport in format PDF sau HTML	27
7.7Urmarierea trimiterilor	27
8Administrarea comenzilor (solicitarea curierurului)	28
8.1Crearea unei comenzi	28
8.2Lansarea sau anulara unei comenzi pentru un punct de lucru	30
8.3Lansarea sau anulara tuturor comenzilor	30
8.4Listare informatii comenzi pe un anumit punct de ridicare	31
8.5Listare informatii comenzi dintr-o anumita perioada	32
9Ridicare din alta locatie	33
.....	34
10Urmarire rambursuri in cont	34

10.1	Lista rambursurilor dintr-un anumit interval de timp.....	34
10.2	Listare rambursuri dupa o anumita data.....	35
10.3	Listare ramburs dupa un anumit cod de bara	35
11	Facturi.....	36
11.1	Lista facturilor.....	36
11.2	Tiparirea facturilor in format PDF.....	36
12	FLUX DE LUCRU PENTRU INTEGRARE.....	38
13	ANEXA.....	38

1 Inregistrare la API StandardUrgentOnlineAPI

1. Trebuie sa va creati un cont. SIGN IN, apoi SIGN UP:

The screenshot shows the UrgentCargus API developer portal. At the top, there is a dark blue header with the text "UrgentCargus API" and a navigation menu with links for "HOME", "APIS", "PRODUCTS", "APPLICATIONS", and "ISSUES". A "SIGN IN" link is located in the top right corner. Below the header is a large blue banner with the text "Welcome to the developer portal!" and two paragraphs of introductory text. A green "Sign up" button is positioned below the text. The main content area is divided into two columns: "API Documentation" and "Developer Support". The "API Documentation" section contains text about automatically generated documentation and the API Console. The "Developer Support" section contains text about engaging with the API community, a blog, issues, and an application gallery. At the bottom right of the page, it says "powered by Microsoft Azure".

UrgentCargus API

HOME APIS PRODUCTS APPLICATIONS ISSUES SIGN IN

Welcome to the developer portal!

Administrators can manage the APIs and customize the content in this portal. [Learn more](#)

Developers can discover and learn about UrgentCargus API. Just sign up for an API key and start consuming this sample API right away!

[Sign up](#)

API Documentation

Check out the automatically generated [API Documentation](#) that describes how to use the APIs and includes code samples in multiple languages. The API Console allows you to directly interact with the API right here in the developer portal.

Developer Support

API publishers can engage directly with their API community, keeping them up to date via the integrated [blog](#). Developers can log and discuss [issues](#) and even submit their applications to the [application gallery](#).

powered by [Microsoft Azure](#)

Sign in

Not a member yet? [Sign up now](#)

Sign in with your username and password

If you are an Administrator you must sign in [here](#).

Email

Password

Remember me on this computer


[Forgot your password?](#)

[Sign in](#)

2. Completati datele necesare apoi confirmati inregistrarea la serviciu apasand link-ul de confirmare primit pe email.

3. Logat, va intoarceti la pagina initiala si accesati PRODUCTS

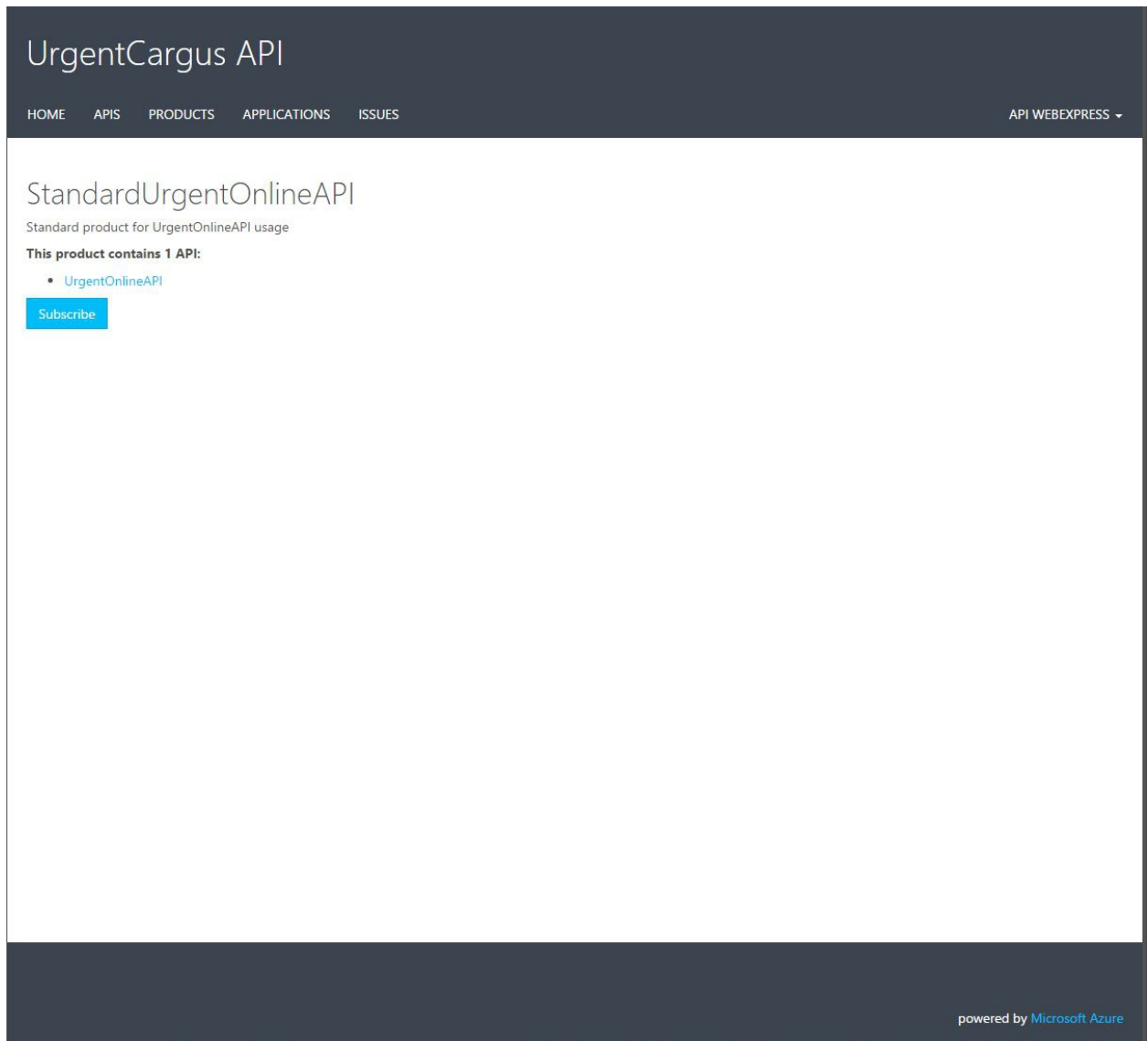
Products

StandardUrgentOnlineAPI

Standard product for UrgentOnlineAPI usage

4. Va inregistrati la API-ul nostru: **StandardUrgentOnlineAPI**



The screenshot shows the 'UrgentCargus API' website. The header is dark grey with the title 'UrgentCargus API' in white. Below the header is a navigation menu with links for 'HOME', 'APIS', 'PRODUCTS', 'APPLICATIONS', and 'ISSUES'. On the right side of the header, there is a link for 'API WEBEXPRESS' with a dropdown arrow. The main content area is white and features the title 'StandardUrgentOnlineAPI' in a large font. Below the title, it says 'Standard product for UrgentOnlineAPI usage'. A section titled 'This product contains 1 API:' lists 'UrgentOnlineAPI' with a blue bullet point. Below the list is a blue 'Subscribe' button. At the bottom right of the page, it says 'powered by Microsoft Azure'.

5. Un administrator de serviciu va aproba cererea de subscriptie. Dupa aprobare veti primi un email de confirmare, moment din care puteti incepe sa consumati serviciul API.

- Accesati PRODUCTS -> **StandardUrgentOnlineAPI** -> **StandardUrgentOnlineAPI** si retineti Primary Key pentru apelul metodelor din API.

The screenshot displays the 'UrgentCargus API' user interface. At the top, there is a navigation bar with links for HOME, APIS, PRODUCTS, APPLICATIONS, and ISSUES, and a user menu for API WEBEXPRESS. The main content area is divided into three sections:

- Profile:** Shows user details: Email (api@webexpress.cargus.ro), First name (API), and Last name (WebExpress). There are buttons for 'Change password' and 'Change account information'.
- Your subscriptions:** Features an 'Analytics reports' button and a table of subscriptions. The table has columns for 'Subscription details', 'Product', and 'State'. One subscription is listed for 'StandardUrgentOnlineAPI' with a state of 'Active'. It includes fields for 'Started on' (8/3/2015), 'Primary key', and 'Secondary key', each with a 'Show | Regenerate' link. A 'Cancel' button is also present.
- Your applications:** Includes a '+ Register application' button and a table with columns for 'Name', 'Category', and 'State'. The table currently shows 'No results found.' Below this, there is a link 'Looking to close your account?' and a 'Close account' button.

At the bottom right of the page, it says 'powered by Microsoft Azure'.

Valoarea de forma **XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX** se va trimite in headerul fiecarui request, sub forma a doi parametri: **'Ocp-Apim-Subscription-Key: XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'**, **'Ocp-Apim-Trace:true'**

- De asemenea, pentru autentificarea la API, este nevoie sa trimiteti in header inca un parametru, de forma: 'Authorization: Bearer TOKEN', unde TOKEN este sirul de caractere obtinut in urma apelului catre metoda **LoginUser**.
- Pentru o definitie completa a API-ului si apelare on-page se poate folosi interfata, accesand APIS->UrgentOnlineAPI:

APIs

Search APIs 

[UrgentOnlineAPI](#)

UrgentCargus API

HOME APIS PRODUCTS APPLICATIONS ISSUES

API WEBEXPRESS ▾

GET AwbDocuments_Get

POST AwbPickup_Post

DELETE Awbs_Delete

GET Awbs_Get

POST Awbs_Post

GET AwbTrace_Get

GET Countries_Get

GET Countries_Get

GET Localities_Get

POST LoginUser_Post

GET OrderDocuments_Get

GET Orders_Get - GET 1

GET Orders_Get - GET 2

PUT Orders_Put

GET PickupLocations_Get

GET PriceTables_Get

POST ShippingCalculation_Post

GET Streets_Get

GET TokenVerification_GET

UrgentOnlineAPI

AwbDocuments_Get

Print awb labels Print awb labels

Try it

Request URL

`https://urgentcargus.azure-api.net/api/AwbDocuments?barCode={barCode}`

Request parameters

barCode string

Request headers

Ocp-Apim-Subscription-Key string Subscription key which provides access to this API. Found in your [Profile](#).

Response 200

OK

application/json text/json application/xml text/xml

Code samples

Curl C# Java JavaScript ObjC PHP Python Ruby

@ECHO OFF

```
curl -v -X GET "https://urgentcargus.azure-api.net/api/AwbDocuments?barCode={barCode}"  
-H "Ocp-Apim-Subscription-Key: {subscription key}"  
--data-ascii "{body}"
```

2 Exemplu de apel PHP:

```
<?php
class UrgentCurier
{
    private $Curl;
    public $url='https://urgentcargus.azure-api.net/api';

    function __construct()
    {
        $this->Curl = curl_init();
        curl_setopt($this->Curl, CURLOPT_RETURNTRANSFER, true);
        curl_setopt($this->Curl, CURLOPT_CONNECTTIMEOUT, 2);
        curl_setopt($this->Curl, CURLOPT_TIMEOUT, 3);
    }

    function CallMethod($function, $parameters="", $verb,$token=null)
    {
        curl_setopt($this->Curl, CURLOPT_POSTFIELDS, $parameters);
        curl_setopt($this->Curl, CURLOPT_CUSTOMREQUEST, $verb);
        curl_setopt($this->Curl, CURLOPT_URL, $this->url . '/' . $function);

        //LoginUser este singura metoda pentru care nu se trimite Token
        if ($function=="LoginUser") {
            curl_setopt($this->Curl, CURLOPT_HTTPHEADER, array('Ocp-Apim-
            Subscription-Key:xxxxxxxxxxxxxxxxxxxxxxxxxxxxx','Ocp-Apim-
            Trace:true','Content-Type: application/json', 'Content-Length: ' .
            strlen($parameters)));
        }
        else {
            curl_setopt($this->Curl, CURLOPT_HTTPHEADER, array('Ocp-Apim-
            Subscription-Key: xxxxxxxxxxxxxxxxxxxxxxxxxxxxx','Ocp-Apim-
            Trace:true','Authorization: Bearer ' . $token, 'Content-Type:
            application/json', 'Content-Length: ' . strlen($parameters)));
        }

        $result=curl_exec($this->Curl);
        $header=curl_getinfo($this->Curl);

        $output['message']=$result;
        $output['status']=$header['http_code'];
        return $output;
    }
}
$urgent = new UrgentCurier();
?>
```

3 Autentificare

3.1 Logare si obtinere token

Metoda este **LoginUser** apelata prin POST si este utilizata la autentificare si obtinere token.

Sunt trimise :

UserName – userul din webexpress
Password – parola corespunzatoare userului

Returneaza :

Token – folosit in continuare in toate metodele apelate ca o masura de autentificare

Exemplu:

```
$fields=array('UserName' => 'test.integration9', 'Password'=> 'a');  
$json=json_encode($fields);  
$login=$urgent->CallMethod('LoginUser', $json, 'POST');  
if ($login['status'] != "200") {  
  
    echo "<b class='bad'>LoginUser: FALSE</b>";  
}  
else {  
    $token = json_decode($login['message']);  
    echo "<b class='good'>LoginUser: </b>".$token;  
}
```

3.2 Verificarea tokenului

Metoda utilizata este **TokenVerification**. Metoda este de tip GET si este folosita daca se doreste verificarea autentificarii.

Sunt trimise:

Token – returnat din metoda LoginUser

Returneaza :

true / false

Exemplu:

```
$result=$urgent->CallMethod('TokenVerification', $json="", 'GET', $token);  
if ($result['status']!="200") {  
    echo "<b class='bad'>Token check: FALSE</b>";  
}  
else{  
    $data=$result['message'];  
    echo "<b class='good'>Token check: </b>".$data; }  
}
```

4 Geografie

4.1 Lista cu tarile din sistem

Metoda utilizata este **Countries** apelata prin GET si intoarce lista tarilor din sistemul Urgent Cargus.

Sunt trimise:

Returneaza:

O lista cu tarile acceptate. Lista cuprinde urmatoarele informatii:
CountryId – id-ul tarii
CountryName – numele tarii
Abbreviation – codul tarii

Exemplu:

```
$resultCountries = $urgent->CallMethod('Countries', $json="", 'GET', $token);  
  
if ($resultCountries['status'] != "200") {  
    echo "<b class='bad'>GET Countries: </b>". $resultCountries['message'];  
}  
else {  
    $jsonCountriesList = $resultCountries['message'];  
    echo "<b class='good'>GET Countries: </b>". $jsonCountriesList;  
}
```

4.2 Lista judetelor dintr-o tara data

Metoda **Counties** este utilizata pentru a returna lista judetelor dintr-o tara. Este o metoda de tip GET, iar in URL este trimis id-ul tarii.

Sunt trimise :

countryId - id-ul tarii returnat in metoda Countries

Returneaza:

O lista cu judetele acceptate. Lista cuprinde urmatoarele informatii:
CountyId – id-ul judetului
Name – numele judetului
Abbreviation – codul judetului

Exemplu:

```
$resultCounties = $urgent->CallMethod('Counties?countryId=1', $json="", 'GET',  
$token);  
if ($resultCounties['status']!="200") {  
    echo "<b class='bad'>GET Counties: </b>". $resultCounties['message'];  
}  
else {  
    $jsonCountiesList = $resultCounties['message'];  
    echo "<b class='good'>GET Counties List: </b>". $jsonCountiesList;  
}
```

4.3 Lista localitatilor dintr-un judet

Metoda **Localities**, apelata prin GET, este utilizata pentru a returna lista localitatilor dintr-un judet dat.

Sunt trimise :

countryId - id-ul tarii returnat in metoda Countries
countyId – id-ul judetului returnat din metoda Counties

Returneaza:

O lista cu localitatile acceptate si informatile aferente. Lista cuprinde urmatoarele informatii:

LocalityId – id-ul localitatii
Name – numele localitatii
ParentId – id-ul agentiei teritoriale de care apartine localitatea
ParentName – numele agentiei teritoriale
ExtraKm – taxabili kilometrii suplimentari
InNetwork – in retea

CountyId – id-ul judetului
CountryId – id-ul tarii
PostalCode – codul postal
MaxHour – ora de ridicare

Exemplu:

```
$resultLocalities = $urgent->CallMethod('Localities?countryId=1&countyId=5', $json="", 'GET', $token);  
if ($resultLocalities ['status']!="200") {  
    echo "<b class='bad'>GET Localities: </b>". $resultLocalities['message'];  
}  
else {  
    $jsonLocalitiesList = $resultLocalities['message'];  
    echo "<b class='good'>GET Localities List: </b>". $jsonLocalitiesList;  
}
```

4.4 Lista strazilor dintr-o localitate

Metoda **Streets**, apelata prin GET, este utilizata pentru a returna lista strazilor din nomenclatorul Urgent Cargus dintr-o localitate data.

Sunt trimise :

localityId - id-ul localitatii returnat in metoda localities.

Returneaza:

O lista cu strazile din nomenclator si informatiile aferente. Lista cuprinde urmatoarele informatii:

StreetId – id-ul strazii
Name – numele strazii

Exemplu:

```
$resultStreets = $urgent->CallMethod('Streets?localityId=157', $json="", 'GET', $token);  
if ($resultStreets ['status']!="200") {  
    echo "<b class='bad'>GET Streets: </b>". $resultStreets['message'];  
}  
else {  
    $jsonStreetsList = $resultStreets['message'];  
    echo "<b class='good'>GET Streets List: </b>". $jsonStreetsList;  
}
```

5 Punctele de ridicare

Un client are mai multe puncte de ridicare. Un user are acces la mai unul sau mai multe puncte de ridicare.

Implicit, un client are un punct de ridicare pe SEDIU care este folosit in aplicatie cu LocationId=0.

5.1 Lista punctelor de ridicare pe client

Metoda **PickupLocations/GetForClient** este de tip GET. Este utilizata pentru a returna lista punctelor de ridicare pentru client.

Sunt trimise :

token – identifica clientul

Returneaza:

O lista cu punctele de ridicare si informatiile aferente. Lista cuprinde urmatoarele informatii:

LocationId – id-ul punctului de ridicare
Name – numele punctului de ridicare
CountyId – id-ul judetului
CountyName – numele judetului
LocalityId – id-ul localitatii
LocalityName – numele localitatii
StreetId – id-ul strazii
StreetName – numele strazii
BuildingNumber – numarul strazii
AddressText - adresa
ContactPerson – persoana de contact
PhoneNumber – numarul persoanei de contact
Email – emailul
AutomaticEOD – ora de inchidere comanda,

Exemplu:

```
$resultPickupLocations = $urgent->CallMethod('PickupLocations/GetForClient', $json="",  
'GET', $token);  
if ($resultPickupLocations ['status']!="200") {  
    echo "<b class='bad'>GET PickupLocations: </b>".  
    $resultPickupLocations['message'];  
}  
else {  
    $jsonPickupLocations = $resultPickupLocations['message'];  
    echo "<b class='good'>GET PickupLocations List: </b>".$jsonPickupLocations;  
}
```

5.2 Lista partenerilor Collect Shop

Metoda **PickupLocations/GetClickCollect** este de tip GET. Este utilizata pentru a returna lista partenerilor Collect Shop.

Sunt trimise :

token – identifica clientul

Returneaza:

O lista cu partenerii si informatiile aferente. Lista cuprinde urmatoarele informatii:

MinihubId – zona de livrare. Trebuie folosit pe printul intern;
Active – daca mai este sau nu partener;
LocationId – id-ul partenerului. Se trimite in LocationId din Recipient;
Name – numele partenerului;
CountyId – id-ul judetului
CountyName – numele judetului
LocalityId – id-ul localitatii
LocalityName – numele localitatii
StreetId – id-ul strazii
StreetName – numele strazii
BuildingNumber – numarul strazii
AddressText - adresa
ContactPerson – persoana de contact
PhoneNumber – numarul persoanei de contact
Email – emailul

O lista a partenerilor se poate afla si prin sistemul celor de la Click & Collect: Harta si API:

<https://clickcollect.github.io/>

Exemplu:

```
$resultPickupLocations = $urgent->CallMethod('PickupLocations/GetClickCollect',
$json="", 'GET', $token);
if ($resultPickupLocations ['status']!="200") {
    echo "<b class='bad'>GET PickupLocations: </b>".
$resultPickupLocations['message'];
}
else {
    $jsonPickupLocations = $resultPickupLocations['message'];
    echo "<b class='good'>GET PickupLocations List: </b>".$jsonPickupLocations;
}
}
```

5.3 Activarea (asignarea) unui punct de ridicare pe user

Metoda **PickupLocations/AssignToUser**, apelata prin POST, este utilizata pentru a adauga(asigna) un punct de ridicare existent pe user.

Sunt trimise :

token – identifica clientul

LocationId – id-ul punctului de ridicare ce se dorese a fi asignat pe utilizatorul logat

Returneaza:

1 – daca a fost activat / eroare – daca nu s-a putut face activarea

Exemplu:

```
$resultAssignToUser = $urgent->CallMethod('PickupLocations/AssignToUser?
LocationId=201008098', $json="", 'POST', $token);

if ($resultAssignToUser ['status']!="200") {
    echo "<b class='bad'>GET AssignToUser: </b>".$resultAssignToUser['message'];
}
else {
    $jsonAssignToUser = $resultAssignToUser['message'];
    echo "<b class='good'>POST AssignToUser Status: </b>".$jsonAssignToUser;
}
}
```

5.4 Listarea punctelor de ridicare active pe user

Metoda **PickupLocations**, apelata prin GET, este utilizata pentru a afisa punctele de ridicare active pe user.

Sunt trimise :

token – identifica clientul

Returneaza:

LocationId – id-ul punctului de ridicare
Name – numele punctului de ridicare
CountyId – id-ul judetului
CountyName – numele judetului
LocalityId – id-ul localitatii
LocalityName – numele localitatii
StreetId – id-ul strazii
StreetName – numele strazii
BuildingNumber – numarul strazii
AddressText - adresa
ContactPerson – persoana de contact
PhoneNumber – numarul persoanei de contact
Email – emailul
AutomaticEOD – ora de inchidere comanda,

Exemplu:

```
$resultPickupLocations = $urgent->CallMethod('PickupLocations', $json="", 'GET',  
$token);  
if ($resultPickupLocations ['status']!="200") {  
    echo "<b class='bad'>GET PickupLocations: </b>".  
    $resultPickupLocations['message'];  
}  
else {  
    $jsonPickupLocations = $resultPickupLocations['message'];  
    echo "<b class='good'>GET PickupLocations Status: </b>".  
    $jsonPickupLocations;  
}
```

5.5 Adaugarea unui punct de ridicare pe un utilizator

Metoda **PickupLocations** este utilizata pentru a adauga un nou punct de ridicare pe user. Punctul de ridicare este implicit activ. Metoda este de tip POST.

Sunt trimise :

AutomaticEOD – ora de inchidere comanda,
LocationId – id-ul punctului de ridicare
Name – numele punctului de ridicare
CountyId – id-ul judetului,
CountyName –numele judetului
LocalityId – id-ul localitatii,
LocalityName – numele localitatii
StreetId - id-ul strazii sau 0
StreetName – numele strazii,
BuildingNumber – numarul strazii,
AddressText - adresa strazii,
ContactPerson – numele persoanei de contact
PhoneNumber – numarul persoanei de contact
Email – emailul

Returneaza

Id-ul punctului de lucru

Exemplu:

```
$jsonPickupLocations = '{  
    "AutomaticEOD": "17:30",
```

```

        "LocationId": "",
        "Name": "Punct ridicare 2 Pitesti",
        "CountyId": 5,
        "CountyName": "Arges",
        "LocalityId": 157,
        "LocalityName": "Pitesti",
        "StreetId": 0,
        "StreetName": "Gheorghe S. Ioan, general, Strada",
        "BuildingNumber": "5",
        "AddressText": "Str. Ionescu Gheorghe",
        "ContactPerson": "Gheorghe",
        "PhoneNumber": "072769821",
        "Email": "a@a.com"
    };

$resultPickupLocationsPost = $urgent->CallMethod('PickupLocations',
$jsonPickupLocations, 'POST', $token);
if ($resultPickupLocationsPost ['status']!="200") {
    echo "<b class='bad'>Post PickupLocations: </b>".
$resultPickupLocationsPost['message'];
}
else {
    $jsonPickupLocationsPost = $resultPickupLocationsPost['message'];
    echo "<b class='good'>Post PickupLocations id-location: </b>".
$jsonPickupLocationsPost;
}

```

5.6 Modificarea unui punct de ridicare

Metoda **PickupLocations** este utilizata pentru a modifica un punct de ridicare. Metoda este de tip PUT.

Sunt trimise :

AutomaticEOD – ora de inchidere comanda, daca nu se doreste se trece NULL
 LocationId – id-ul punctului de ridicare
 Name – numele punctului de ridicare
 CountyId – id-ul judetului,
 CountyName –numele judetului,
 LocalityId – id-ul localitatii,
 LocalityName – numele localitatii,
 StreetId - id-ul strazii sau 0
 StreetName – numele strazii,
 BuildingNumber – numarul strazii,
 AddressText - adresa punctului,
 ContactPerson – numele persoanei de contact
 PhoneNumber – numarul persoanei de contact
 Email – emailul

Returneaza

True

Exemplu:

```

$jsonPickupLocations =
{"AutomaticEOD":null,
 "LocationId": "31498",
 "Name": "ecomMODIFICAT",
 "CountyId": 5,
 "CountyName": "Arges",
 "LocalityId": 157,
 "LocalityName": "pitesti",
 "StreetId": 0,
 "StreetName": "Gheorghe S. Ioan, general, Strada",
 "BuildingNumber": "5",
 "AddressText": "bl 5 ap 5",
 "ContactPerson": "Gheorghe",
 "PhoneNumber": "072769821",
 "Email": "a@a.com"
};

$result = $urgent->CallMethod('PickupLocations', $jsonPickupLocations, 'PUT',
$token);

if ($result['status']!="200") {
    echo("<b class='bad'>PUT PickupLocations: </b>". $result['message']);
}
else {
    $data=$result['message']; echo "<b class='good'>PUT PickupLocations:
</b>". $data;
}

```

6 Tarif

6.1 Lista tarifelor din contract

Metoda **PriceTable** , apelata prin GET, este utilizata pentru afisa tarifele stabilite in contract.

Sunt trimise :

token – identifica clientul

Returneaza:

PriceTableId – id-ul tarifului,
Name – numele tarifului

Exemplu:

```

$resultPriceTables =$urgent->CallMethod('PriceTables', $json="", 'GET', $token);

if ($resultPriceTables ['status']!="200") {
    echo"<b class='bad'>GET PriceTables : </b>". $resultPriceTables ['message'];
}
else {
    $data=$resultPriceTables ['message'];
    echo "<b class='good'>GET PriceTables : </b>". $data;
}

```

6.2 Calcul tarif pentru o expediere

Metoda **ShippingCalculations**, apelata prin POST, este utilizata pentru a afla tariful unui transport .

Sunt trimise :

FromLocalityId – id-ul localitatii expeditorului,
ToLocalityId - id-ul localitatii destinatarului,
FromCountyName - numele judetului expeditorului / OPTIONAL,
FromLocalityName – numele localitatii expeditorului/OPTIONAL
ToCountyName – numele judetului destinatarului/ OPTIONAL
ToLocalityName – numele localitatii destinatarului/OPTIONAL
Parcels – numarul de colete
Envelopes - numarul de plicuri/ SE POATE PUNE PANA LA 9
TotalWeight – greutatea totala a expedierii
ServiceId – id-ul serviciu 1 – standard, 4 partener
DeclaredValue – valoarea declarata
CashRepayment - suma de ramburs ce se intoarce in plic
BankRepayment – suma de ramburs ce se vireaza in cont
OtherRepayment – orice se doreste retur// confirmare de primire, colet la schimb
PaymentInstrumentId – id-ul instrumentului de plata// 1 – cec, 2 –bo, 3
PaymentInstrumentValue – valoarea instrumentului de plata
OpenPackage – deschidere colet// se foloseste TRUE sau FALSE
ShipmentPayer – platitorul transportului. Se trimite 1 pentru expeditor si 2 pentru destinatar
PriceTableId – id-ul tarifului returnat din metoda PriceTables

Returneaza

BaseCost - tariful de baza fara contract
ExtraKmCost – tariful kilometrilor suplimentari
WeightCost – tariful din contract
InsuranceCost – tariful pentru valoarea declarata (asigurare)
SpecialCost – taxe speciale
RepaymentCost – tarif pentru ramburs : numerar(plic) sau retur (OtherRepayment)
Subtotal – total fara TVA
Tax - TVA
GrandTotal – Total tarif

Exemplu:

```
$json='{
  "FromLocalityId" :150,
  "ToLocalityId" :150,
  "Parcels": 1,
  "Envelopes": 0,
  "TotalWeight": 1,
  "DeclaredValue": 0,
  "CashRepayment": 0,
  "BankRepayment": 0,
  "OtherRepayment": "",
  "OpenPackage": true,
  "PriceTableId": 23049,
  "ShipmentPayer": 1,
}';
$result=$urgent->CallMethod('ShippingCalculation', $json, 'POST', $token);
if ($result['status']!="200") {
  echo("<b class='bad'>POST ShippingCalculation: </b>".$result['message']);
}
```

```

else {
    $data=$result['message'];
    echo "<b class='good'>POST ShippingCalculation: </b>".$data;
}

```

7 Administrarea notelor de transport

7.1 Generarea notelor de transport

Metoda **Awbs**, apelata prin POST, este utilizata pentru a adauga un nou awb.

Recomandam ca inaintea inceperii generarii de AWB-uri intr-o anumita zi, sa se deschida o comanda noua, utilizand metoda **Orders** (pag. 31).

Sunt trimise :

SenderClientId – Pentru toate cazurile este null. Este o facilitate creata pentru cei care vor sa creeze AWB-uri in numele altui client.

TertiaryClientId – id-ul tertului, daca se face trimiterea in contul altui client. Se va tarifa cu tariful clientului asociat tarifului.

Sender - Expeditorul

LocationId – id-ul punctului de ridicare // 0 pentru sediu

Recipient – Destinatarul

Name – numele destinatarului

CountyId – id-ul judetului

CountyName – judetul // Obligatziu daca nu se trimite LocalityId

LocalityId – id-ul localitatii // Obligatziu daca nu sunt trimise CountyName impreuna cu LocalityName

LocalityName – numele localitatii // Obligatziu daca nu se trimite LocalityId

StreetId – id-ul strazii // Optional

StreetName – numele strazii // Optional

BuildingNumber – numarul strazii

AddressText – detalii

ContactPerson – persoana de contact

PhoneNumber – numarul de telefon

Email – email-ul destinatarului

Parcels – numarul de colete

Envelopes – numarul de plicuri. Se pot trimite maxim 9

TotalWeight – greutate expediere aproximativa

DeclaredValue – valoarea declarata pentru asigurare

CashRepayment – ramburs numerar (suma ramburs care se intoarce la expeditor se face in plic)

BankRepayment – ramburs in cont (suma ramburs care se intoarce la expeditor in cont colector)

OtherRepayment – alte retururi

OpenPackage – deschidere colet

HasTertReimbursement – pentru 'true' rambursul se intoarce la TertiaryClientId, care trebuie trimis obligatziu

PriceTableId – id-ul tarifului

ShipmentPayer – platitorul transportului. Se trimite 1 pentru expeditor si 2 pentru destinatar

SaturdayDelivery – livrare sambata

MorningDelivery - livrare matinala

Observations - observatii

PackageContent – continutul expedierii

CustomString – referinta Serie Client

SenderReference1 – referinta expeditor 1
RecipientReference1 – referinta destinatar 1
RecipientReference2 – referinta destinatar 2
InvoiceReference – referinta facturare

Returneaza:

Numarul codului de bara generat

Exemplu:

```
$json='{
  "Sender":{"LocationId":28244},
  "Recipient":{
    "Name":"Ion Popescu",
    "CountyId":33,
    "LocalityId":161,
    "StreetId":0,
    "StreetName":"Str. Ionescu",
    "BuildingNumber":"25",
    "AddressText":"Bl. 129, Sc. 3, Ap. 4",
    "ContactPerson":" Ion Popescu ",
    "PhoneNumber":"072111222",
    "Email":"a@a.com"
  },
  "Parcels":1,
  "Envelopes":0,
  "TotalWeight":1,
  "DeclaredValue":0,
  "CashRepayment":0,
  "BankRepayment":0,
  "OtherRepayment":"","
  "OpenPackage":true,
  "PriceTableId":0,
  "ShipmentPayer":1,
  "SaturdayDelivery":true,
  "Observations":"Observatii",
  "PackageContent":"Colet",
  "CustomString":"Serie client",
  "SenderReference1": "Ref 1 exp",
  "RecipientReference1": "Ref 1 dest",
  "RecipientReference2": "Ref 2 dest",
  "InvoiceReference": "referinta facturare"
}';

$result=$urgent->CallMethod('Awbs', $json, 'POST', $token);
if ($result['status']!="200") {
  echo("<b class='bad'>POST Awb: </b>". $result['message']);
}
else {
  $data=$result['message'];
  echo "<b class='good'>POST Awb: </b>". $data;
}
```

7.2 Generarea notelor de transport in sistem Collect Shop

Metoda **Awbs**, apelata prin POST, este utilizata pentru a adauga un nou awb.

Sunt trimise :

SenderClientId – Pentru toate cazurile este null. Este o facilitate creata pentru cei care vor sa creeze AWB-uri in numele altui client.

TertiaryClientId – id-ul tertului, daca se face trimiterea in contul altui client. Se va tarifa cu tariful clientului asociat tarifului.

Sender - Expeditorul

LocationId – id-ul punctului de ridicare // 0 pentru sediu

Recipient – Destinatarul

LocationId – id-ul partenerului Collect Shop

ContactPerson – destinatarul ce trebuie sa primeasca coletul

PhoneNumber –telefonul destinatarului

Email – email-ul destinatarului

Parcels – numarul de colete

Envelopes – numarul de plicuri. Se pot trimite maxim 9

TotalWeight – greutate expediere aproximativa

DeclaredValue – valoarea declarata pentru asigurare

CashRepayment – ramburs numerar (suma ramburs care se intoarce la expeditor se face in plic)

BankRepayment – ramburs in cont (suma ramburs care se intoarce la expeditor in cont colector)

OtherRepayment – alte returnuri

OpenPackage – deschidere colet

HasTertReimbursement – nu se trimite

PriceTableId – id-ul tarifului

ShipmentPayer – platitorul transportului. Se trimite doar 1 (expeditor)

ServiceId – id-ul serviciului (30)

SaturdayDelivery – livrare sambata

MorningDelivery - livrare matinala

Observations - observatii

PackageContent – continutul expedierii

CustomString – referinta Serie Client

SenderReference1 – referinta expeditor 1

RecipientReference1 – referinta destinatar 1

RecipientReference2 – referinta destinatar 2

InvoiceReference – referinta facturare

Returneaza:

Numarul codului de bara generat

7.3 Stergerea unei note de transport

Metoda **Awbs** sterge o nota de transport pentru care nu exista un checkpoint. Este o metoda de tip DELETE.

Sunt trimise: barCode – numarul notei de transport

Returneaza: true – in caz de success
false – daca nota nu exista sau are checkpoint

Exemplu:

```
$resultDelete=$urgent->CallMethod('Awbs?barCode= 804713464', $json="", 'DELETE',  
$token);  
if ($resultDelete ['status']!="200") {
```

```

        echo("<b class='bad'>Awb deleted : </b>".$resultDelete ['message']);
    }
    else {
        $data=$resultDelete ['message'];
    }
}

```

7.4 Listarea notelor dintr-o anumita perioada

Metoda **AWBS/ GETBYDATE**, apelata prin GET, este utilizata pentru a returna lista awb-urilor dintr-o anumita perioada.

Sunt trimise :

FromDate - data inceput in format ll-zz-aaaa
 ToDate - data sfarsit in format ll-zz-aaaa
 pageNumber – numarul paginii afisat
 itemsPerPage – numarul awb-urilor pe pagina

Returneaza:

O lista cu awb-urile. Lista cuprinde urmatoarele informatii:

```

[
  {
    "SenderClientId": null,
    "TertiaryClientId": null,
    "Sender": {
      "LocationId": 1005962049,
      "Name": "ECOM TEST",
      "CountyId": 0,
      "CountyName": "Ilfov",
      "LocalityId": 29445916,
      "LocalityName": "Tunari",
      "StreetId": 0,
      "StreetName": "",
      "BuildingNumber": "",
      "AddressText": "nr: 32; Sos centura nr 32",
      "ContactPerson": "Ionut Petraru",
      "PhoneNumber": "",
      "Email": ""
    },
    "Recipient": {
      "LocationId": 1,
      "Name": "test",
      "CountyId": 1,
      "CountyName": "Bucuresti",
      "LocalityId": 150,
      "LocalityName": "BUCURESTI",
      "StreetId": 0,
      "StreetName": "",
      "BuildingNumber": "",
      "AddressText": "dsgfggfghgfhg",
      "ContactPerson": "",
      "PhoneNumber": "",
      "Email": ""
    },
    "Parcels": 0,
    "Envelopes": 1,
    "TotalWeight": 1,
    "ServiceId": null,
    "DeclaredValue": 0,
  }
]

```



```

    "CashRepayment": 120,
    "BankRepayment": 0,
    "OtherRepayment": "",
    "PaymentInstrumentId": 0,
    "PaymentInstrumentValue": 0,
    "HasTertReimbursement": false,
    "OpenPackage": false,
    "PriceTableId": 17430,
    "ShipmentPayer": 3,
    "SaturdayDelivery": false,
    "MorningDelivery": false,
    "Observations": "",
    "PackageContent": "",
    "CustomString": "",
    "BarCode": "804523201",
    "ParcelCodes": null,
    "ValidationDate": "2016-02-08T06:38:06.267",
    "ShippingCost": null,
    "Status": "Tiparit",
    "SenderReference1": "",
    "RecipientReference1": "",
    "RecipientReference2": "",
    "InvoiceReference": ""
  ]

```

Exemplu:

```

$resultGetByDate=$urgent->CallMethod('Awbs/GetByDate?FromDate=02-08-2016&ToDate=02-08-2016&pageNumber=1&itemsPerPage=100', $json="", 'GET', $token);

if ($resultGetByDate['status']!="200") {
    echo "<b class='bad'>GET GetByDate : </b>".$resultGetByDate['message'];
}
else {
    $dataGetByDate=$resultGetByDate['message'];
    echo "<b class='good'>GET GetByDate : </b>".$dataGetByDate;
}

```

7.5 Listare informatii note transport pentru o anumita comanda

Metoda **Awbs**, apelata prin GET, intoarce detalii despre notele trimise.

Sunt trimise:

barCode – awb-ul cautat
sau
orderId – numarul comenzii
Atentie! Nu se trimit impreuna.

Returneaza:

```

[
  {
    "SenderClientId": null,
    "TertiaryClientId": null,
    "Sender": {
      "LocationId": 1005962049,
      "Name": "ECOM TEST",
      "CountyId": 0,
      "CountyName": "Ilfov",
      "LocalityId": 29445916,
      "LocalityName": "Tunari",

```

```

"StreetId": 0,
"StreetName": "",
"BuildingNumber": "",
"AddressText": "nr: 32; Sos centura nr 32",
"ContactPerson": "Ionut Petraru",
"PhoneNumber": "",
"Email": ""
},
"Recipient": {
"LocationId": 1,
"Name": "test",
"CountyId": 1,
"CountyName": "Bucuresti",
"LocalityId": 150,
"LocalityName": "BUCURESTI",
"StreetId": 0,
"StreetName": "",
"BuildingNumber": "",
"AddressText": " dsgfggfhgfhg",
"ContactPerson": "",
"PhoneNumber": "",
"Email": ""
},
"Parcels": 0,
"Envelopes": 1,
"TotalWeight": 1,
"ServiceId": null,
"DeclaredValue": 0,
"CashRepayment": 120,
"BankRepayment": 0,
"OtherRepayment": "",
"PaymentInstrumentId": 0,
"PaymentInstrumentValue": 0,
"HasTertReimbursement": false,
"OpenPackage": false,
"PriceTableId": 17430,
"ShipmentPayer": 3,
"SaturdayDelivery": false,
"MorningDelivery": false,
"Observations": "",
"PackageContent": "",
"CustomString": "",
"BarCode": "804523201",
"ParcelCodes": null,
"ValidationDate": "2016-02-08T06:38:06.267",
"ShippingCost": null,
"Status": "Tiparit",
"SenderReference1": "",
"RecipientReference1": "",
"RecipientReference2": "",
"InvoiceReference": ""

```

Exemplu:

Pentru un AWB:

```

$resultAwbDetails=$urgent->CallMethod('Awbs?barCode=804737910', $json="",
'GET', $token);
if ($resultAwbDetails['status']!="200") {

```

```

        echo "<b class='bad'>GET awb details: </b>".
$resultAwbDetails['message'];
    }
    else {
        $data=$resultAwbDetails['message'];
        echo "<b class='good'>GET awb details By AWB: </b>".$data;
    }
}
Pentru o comanda:
$resultAwbDetails=$urgent->CallMethod('Awbs?orderId=201243788', $json="",
'GET', $token);
if ($resultAwbDetails['status']!="200") {
    echo "<b class='bad'>GET awb details: </b>".
$resultAwbDetails['message'];
}
else {
    $data=$resultAwbDetails['message'];
    echo "<b class='good'>GET awb details By order ID : </b>".$data;
}

```

7.6 Tiparire note de transport in format PDF sau HTML

Metoda **AwbDocuments**, apelata prin GET, este utilizata pentru printarea coletelor in PDF sau HTML. Se poate specifica daca este A4 sau Label

Sunt trimise :

barCode - o lista (json) cu awb-urile pentru printare
type – PDF sau HTML
format - 0 - A4; 1 - Label 10x14

Returneaza:

Codul in base64

Exemplu:

```

header("Content-type:application/pdf");

$awb=array(804419419,804418863);
$jsonAwb=json_encode($awb);

$result=$urgent->CallMethod('AwbDocuments?barCodes='.
$jsonAwb.'&type=PDF&format=1', $json="", 'GET', $token);
if ($result['status']!="200") {
    echo("<b class='bad'>Awb Documents: </b>".$result['message']);
}
else {
    $data=$result['message'];
    echo base64_decode($data);
}

```

7.7 Urmarierea trimiturilor

Metoda **AwbTrace**, este de tipul GET, si este utilizata pentru urmarirea coletelor si afisarea statusurilor.

Sunt trimise :

barCode - o lista cu awb-urile cautate

Returneaza:

Info cu evenimete. Lista cuprinde urmatoarele informatii:

Code – numarul notei
Type -
MeasuredWeight - greutatea
VolumetricWeight – greutate volumetrica
Event
Date – data eveniment
Description – descriere eveniment
LocalityName - centru

Exemplu:

```
$awbList=array(804115458, 804346669);  
$jsonAwb=json_encode($awbList);  
$resultTrace = $urgent->CallMethod('AwbTrace?barCode='.$jsonAwb, $json="", 'GET',  
$token);  
if ($resultTrace['status'] != "200") {  
    echo"<b class='bad'>GET ERROR Trace: </b>".$resultTrace['message'];  
}  
else {  
    $result = $resultTrace['message'];  
    echo "<b class='good'>GET Trace: </b>".$result;  
}
```

8 Administrarea comenzilor (solicitarea curierului)

8.1 Crearea unei comenzi

Metoda **Orders**, apelata prin POST, trimite o noua solicitare de curier.

Sunt trimise :

Parcels – numarul de colete
Envelopes – numarul de plicuri
TotalWeight – greutatea totala a expedierilor din comanda
PickupStartDate – data si ora de inceput pentru ridicarea comenzii
PickupEndDate – data si ora de sfarsit pentru ridicarea comenzii
LocationId – id-ul punctului de lucru –
- daca se trimite LocationId, atunci informatiile de mai jos nu se mai trimit.
LocationId cu valoarea 0 este sediul.
- daca LocationId este NULL (sau nu este trimis), atunci se doreste
ridicarea unei comenzi dintr-o alta adresa, decat cea de la un punct de lucru
existent si se trimit datele de mai jos.
Name – numele expeditorului de unde se ridica comanda
CountyId – id-ul judetului
CountyName – numele judetului
LocalityId – id-ul localitatii
LocalityName – numele localitatii
StreetId – id-ul strazi
StreetName – numele strazii exact din nomenclatorul Urgent Cargus
BuildingNumber – numarul strazii
AddressText – adresa comenzii
ContactPerson – persoana de contact pentru comanda curenta
PhoneNumber – numarul de telefon
Email – email-ul

Pentru comenzi din sediu:

```
{
  "parcels":1,
  "envelopes":0,
  "totalWeight":0,
  "pickupStartDate":"2015-11-10T00:00:00.000+0200",
  "pickupEndDate":"2015-11-10T15:00:00.000+0200",
  "name":"Orange Service",
  "locationId":0,
}
```

Pentru comenzi din alt punct de ridicare activ:

```
{
  "parcels":1,
  "envelopes":0,
  "totalWeight":0,
  "pickupStartDate":"2015-11-10T00:00:00.000+0200",
  "pickupEndDate":"2015-11-10T15:00:00.000+0200",
  "name":"Orange Service",
  "locationId":0,
}
```

Pentru comenzi din alta adresa:

```
{
  "parcels":1,
  "envelopes":0,
  "totalWeight":0,
  "pickupStartDate":"2015-11-10T00:00:00.000+0200",
  "pickupEndDate":"2015-11-10T15:00:00.000+0200",
  "name":"Orange Service",
  "countyId":44,
  "countyName":"Bucuresti",
  "localityId":150,
  "localityName":"Bucuresti",
  "streetId":1234,
  "streetName":"Calea Giulesti",
  "buildingNumber":"303",
  "addressText":"Calea Giulesti, nr. 303",
  "contactPerson":"cana",
  "phoneNumber":"0744440587",
  "email":"cristi.cana@orange.com"
}
```

Returneaza:

Numarul comenzii de forma 201272566

Exemplu:

```
$jsonOrder = '{
  "Parcels": 2,
  "Envelopes": 1,
  "TotalWeight": 0,
  "PickupStartDate": "2016-02-16T08:00:35.088Z",
  "PickupEndDate": "2016-02-16T08:00:35.088Z",
  "LocationId": 0,
  "Name": "GAMA COMPANY",
  "CountyId": 0,
```

```

    "CountyName": "Romania",
    "LocalityId": 157,
    "LocalityName": "FOCSANI",
    "StreetId": 0,
    "StreetName": "Nume strada",
    "BuildingNumber": "Numar strada",
    "AddressText": "Adresa",
    "ContactPerson": "Popescu Marius",
    "PhoneNumber": "07111111",
    "Email": "a@a.com"
};

$resultPostOrder = $urgent->CallMethod('Orders', $jsonOrder, 'POST', $token);
if ($resultPostOrder['status']!="200") {
    echo("<b class='bad'>POST Order: </b>".$resultPostOrder['message']);
}
else {
    $dataPostOrder=$resultPostOrder['message'];
    echo "<b class='good'>POST Order: </b>".$dataPostOrder;
}

```

8.2 Lansarea sau anularea unei comenzi pentru un punct de lucru

Metoda **Orders**, apelata prin PUT, valideaza sau anuleaza o comanda de pe un punct de lucru.

Sunt trimise:

LocationId – id-ul punctului de lucru – 0 pentru SEDIU
 action – id-ul actiuni . daca se trimite 0 anuleaza si 1 valideaza
 PickupStartDate – data si ora de inceput pentru ridicarea comenzii
 PickupEndDate – data si ora de sfarsit pentru ridicarea comenzii

Returneaza:

Numarul comenzii

Exemplu:

```

$resultPutOrder = $urgent->CallMethod('Orders?
locationId=201008098&action=0&PickupStartDate=2016-02-
16T12:16:15.827Z&PickupEndDate=2016-02-16T12:19:15.827Z',$json="", 'PUT', $token);
if ($resultPutOrder['status']!="200") {
    echo("<b class='bad'>PUT Order: </b>".$resultPutOrder['message']);
}
else {
    $dataPutOrder=$resultPutOrder['message'];
    echo "<b class='good'>PUT Order: </b>".$dataPutOrder;
}

```

8.3 Lansarea sau anularea tuturor comenzilor

Metoda **Orders/PutAll** valideaza sau anuleaza toate comenzile, pe toate punctele de ridicare.

Sunt trimise:

action – daca se trimite 0 anuleaza si 1 valideaza
 PickupStartDate – data si ora de inceput pentru ridicarea comenzii
 PickupEndDate – data si ora de sfarsit pentru ridicarea comenzii

Returneaza:

Numarul comenzii

Exemplu:

```
$resultPutOrder = $urgent->CallMethod('Orders/PutAll?
action=1&PickupStartDate=2016-02-16T12:16:15.827Z&PickupEndDate=2016-02-
16T12:19:15.827Z',$json="", 'PUT', $token);
if ($resultPutOrder['status']!="200") {
    echo("<b class='bad'>PUT Order: </b>".$resultPutOrder['message']);
}
else {
    $dataPutOrder=$resultPutOrder['message'];
    echo "<b class='good'>PUT Order: </b>".$dataPutOrder;
}
```

8.4 Listare informatii comenzi pe un anumit punct de ridicare

Metoda **Orders** apelata prin GET aduce informatii despre comenzile dintr-un anumit punct de ridicare dat.

Sunt trimise:

LocationId – id-ul punctului de lucru – 0 pentru SEDIU
status – statusul comenzii – 0 comanda curenta / 1 comanda validata
pageNumber – numarul de pagini afisate
itemsPerPage – numarul de comenzii pe pagina

Returneaza:

O lista de comenzii. O comanda contine urmatoarele informatii:

OrderId - numarul comenzi
LocationId - id-ul punctului de lucru
PickUpSiteName - adresa punctului de lucru
PickupStartDate - data si ora de inceput pentru ridicarea comenzii
PickupEndDate - data si ora de sfarsit pentru ridicarea comenzii
- daca comanda are statusul 0 atunci este afisata data de generare a comenzii
NoParcel - numarul total de colete din comanda
NoEnvelop - numarul total de plicuri din comanda
TotalWeight - greutatea totala a expedierilor din comanda
NoAwb - numarul total de awb-uri din comanda
Observations - observatiile din comanda
PackageContent - continut colet
CreationDate - data si ora de creare a comenzii
ValidationDate - data si ora de trimitere a comenzii
ProcessedDate - data procesarii
OrdStatus – statusul comenzii
AdresaExp – adresa expeditorului
Email – ar trebui sa fie adresa de email de pe punctual de lucru
OrderType – tipul de comanda. Prin ce canal a fost generate

Exemplu:

```
$resultOrders = $urgent->CallMethod('Orders?
locationId=0&status=1&pageNumber=1&itemsPerPage=100', $json="", 'GET', $token);

if ($resultOrders['status'] != "200") {
    echo("<b class='bad'>GET Order: </b>".$resultOrders['message']);
}
else {
    $dataOrders=$resultOrders['message'];
}
```

```

        echo "<b class='good'>GET Orders: </b>".$dataOrders;
    }

```

8.5 Listare informatii comenzi dintr-o anumita perioada.

Metoda **Orders/GetByDate** este apelata prin GET si aduce informatii despre comenzile date intr-o anumita perioada.

Sunt trimise:

FromDate - data inceput in format aaaa-ll-zz
 ToDate - data sfarsit in format aaaa-ll-zz
 pageNumber – numarul paginii afisat
 itemsPerPage – numarul awb-urilor pe pagina

Returneaza:

O lista de comenzi. O comanda contine urmatoarele informatii:

OrderId - numarul comenzii
 LocationId - id-ul punctului de lucru
 PickupSiteName - adresa punctului de lucru
 PickupStartDate - data si ora de inceput pentru ridicarea comenzii
 PickupEndDate - data si ora de sfarsit pentru ridicarea comenzii
 daca comanda are statusul 0 atunci este afisata data de generare a comenzii
 NoParcel - numarul total de colete din comanda
 NoEnvelop - numarul total de plicuri din comanda
 TotalWeight - greutatea totala a expedierilor din comanda
 NoAwb - numarul total de awb-uri din comanda
 Observations - observatiile din comanda
 PackageContent - continut colet
 CreationDate - data si ora de creare a comenzii
 ValidationDate - data si ora de trimitere a comenzii
 ProcessedDate - data procesarii
 OrdStatus – statusul comenzii
 AdresaExp – adresa expeditorului
 Email – ar trebui sa fie adresa de email de pe punctual de lucru
 OrderType – tipul de comanda. Prin ce canal au fost generate

Exemplu:

```

$resultOrdersGetByDate =$urgent->CallMethod('Orders/GetByDate?FromDate=2016-02-15&ToDate=2016-02-19&pageNumber=1&itemsPerPage=100', $json="", 'GET', $token);

if ($resultOrdersGetByDate ['status']!="200") {
    echo "<b class='bad'>GET Result Orders ByDate : </b>".
    $resultOrdersGetByDate['message'];
}
else {
    $dataOrdersGetByDate = $resultOrdersGetByDate['message'];
    echo "<b class='good'>GET Result Orders ByDate: </b>".
    $dataOrdersGetByDate;
}

```


9 Ridicare din alta locatie

Metoda AwbPickup , apelata prin POST, este utilizata pentru a genera un nou awb si a transmite o comanda noua catre curier.

PickupStartDate – data inceput comanda
PickupEndDate – data sfarsit comanda
SenderClientId –Este o facilitate creata pentru cei care vor sa creeze AWB-uri in numele altui client.
TertiaryClientId – id-ul tertului, daca se face trimiterea in contul altui client. Se va tarifa cu tariful clientului asociat tarifului.
Sender - Expeditorul
LocationId – id-ul expeditorului,
Name – numele destinatarului
CountyId – id-ul judetului
LocalityId – id-ul localitatii
StreetId – id-ul strazii
StreetName – numele strazii// Optional
BuildingNumber – numarul strazii
AddressText – detalii
ContactPerson – persoana de contact
PhoneNumber – numarul de telefon
Email – email-ul destinatarului
Recipient – Destinatarul
LocationId – id-ul locatiei
Name – numele destinatarului
CountyId – id-ul judetului
LocalityId – id-ul localitatii
StreetId – id-ul strazii
StreetName – numele strazii// Optional
BuildingNumber – numarul strazii
AddressText – detalii
ContactPerson – persoana de contact
PhoneNumber – numarul de telefon
Email – email-ul destinatarului

Parcels – numarul de colete
Envelopes – numarul de plicuri. Se pot trimite maxim 9
TotalWeight – greutate expediere aproximativa
DeclaredValue – valoarea declarata pentru asigurare
CashRepayment – ramburs numerar (suma ramburs care se intoarce la expeditor se face in plic)
BankRepayment – ramburs in cont (suma ramburs care se intoarce la expeditor in cont colector)
OtherRepayment – alte returnari
OpenPackage – deschidere colet
HasTertReimbursement – pentru 'true' rambursul se intoarce la TertiaryClientId, care trebuie trimis obligatoriu
PriceTableId – id-ul tarifului
ShipmentPayer – platitorul transportului. Se trimite 1 pentru expeditor si 2 pentru destinatar
SaturdayDelivery – livrare sambata
MorningDelivery - livrare matinala
Observations - observatii
PackageContent – continutul expedierii
CustomString – referinta Serie Client
SenderReference1 – referinta expeditor 1
RecipientReference1 – referinta destinatar 1

RecipientReference2 – referinta destinatar 2
InvoiceReference – referinta facturare

10 Urmarire rambursuri in cont

Metoda **CashAccount** este utilizata pentru a afisa informatiile despre rambursurile in cont colector.

10.1 Lista rambursurilor dintr-un anumit interval de timp

Metoda **CashAccount/GetByDate** este apelata prin GET si aduce informatii despre rambursuri dintr-o anumita perioada.

Sunt trimise :

FromDate - data inceput in format aaaa-ll-zz
ToDate - data sfarsit in format aaaa-ll-zz
token – identifica clientul

Returneaza:

Date – data de intocmire a awb-ului
BarCode – numarul awb-ului
Sender – expeditorul
Receiver – destinatarul
FromLocality – localitatea expeditorului
ToLocality – localitatea destinatarului
CustomString – serie client
RepaymentDate – data intocmire ramburs
RepaymentID – chitanta de ramburs
RepaymentValue – valoarea rambursului
DeductionDate – data virare OP
Deductionid – numarul ordinului de plata

Exemplu:

```
$resultCashAccount =$urgent->CallMethod('CashAccount/GetByDate?FromDate=2016-03-24&ToDate=2016-03-25', $json="", 'GET', $token);
```

```
if ($resultCashAccount ['status']!="200") {  
    echo "<b class='bad'>GET CashAccount : </b>".$resultCashAccount['message'];  
}  
else {  
    $dataCashAccount =$resultCashAccount['message'];  
    echo "<b class='good'>GET Get Cash Account : </b>".$dataCashAccount;
```

}

10.2 Listare rambursuri dupa o anumita data

Metoda **CashAccount/GetByDeductionDate** este apelata prin GET si aduce informatii despre OP-urile dintr-o anumita data.

Sunt trimise :
DeductionDate – data OP-ului
token – identifica clientul

Returneaza:

Date – data de intocmire a awb-ului
BarCode – numarul awb-ului
Sender – expeditorul
Receiver – destinatarul
FromLocality – localitatea expeditorului
ToLocality – localitatea destinatarului
CustomString – serie client
RepaymentDate – data intocmire ramburs
RepaymentID – chitanta de ramburs
RepaymentValue – valoarea rambursului
DeductionDate – data virare OP
Deductionid – numarul ordinului de plata

Exemplu:

```
$result=$urgent->CallMethod('CashAccount/GetByDeductionDate?
DeductionDate=2016-03-29',$json='', 'GET', $token);

print_r($result);

if ($result['status']!="200") {
    echo "<b class='bad'>GET CashAccount?GetByDeductionDate: </b>".
    $result['message'];
}
else {
    $data=$result['message'];
    echo "<b class='good'>GET CashAccount?GetByDeductionDate: </b>".$data;
}
```

10.3 Listare ramburs dupa un anumit cod de bara

Sunt trimise :
token – identifica clientul
barCode – numarul awb-ului la care se asteapta rambursul

Returneaza:

Date – data de intocmire a awb-ului
BarCode – numarul awb-ului
Sender – expeditorul
Receiver – destinatarul
FromLocality – localitatea expeditorului

ToLocality – localitatea destinatarului
CustomString – serie client
RepaymentDate – data intocmire ramburs
RepaymentID – chitanta de ramburs
RepaymentValue – valoarea rambursului
DeductionDate – data virare OP
Deductionid – numarul ordinului de plata

Exemplu:

```
$resultCashAccount =$urgent->CallMethod('CashAccount?barCode=804373743',  
$json="", 'GET', $token);  
  
if ($resultCashAccount ['status']!="200") {  
    echo "<b class='bad'>GET CashAccount : </b>". $resultCashAccount['message'];  
}  
else {  
    $dataCashAccount =$resultCashAccount['message'];  
    echo "<b class='good'>GET Get Cash Account : </b>". $dataCashAccount;  
}
```

11 Facturi

11.1 Lista facturilor

Metoda **Invoices** intoarce informatii despre facturile dintr-o anumita perioada.

Sunt trimise :

FromDate - data inceput in format aaaa-ll-zz
ToDate - data sfarsit in format aaaa-ll-zz
pageNumber – numarul paginii afisat
itemsPerPage – numarul awb-urilor pe pagina

Returneaza:

InvoiceId - 2109644734,
Date – data emitere
DueDate – data facturare
Series – serie factura
Number – numar facture
Value – total fara tva
Total – total
Closed – facture inchisa
Balance - sold
CurrencyId – valuta

Exemplu:

```
$resultInvoices =$urgent->CallMethod('Invoices?FromDate=2016-01-01&ToDate=2016-  
02-19&pageNumber=1&itemsPerPage=100', $json="", 'GET', $token);  
if ($resultInvoices ['status']!="200") {  
    echo "<b class='bad'>GET Invoices : </b>". $resultInvoices['message'];  
} else {  
    $dataInvoices =$resultInvoices['message'];  
    echo "<b class='good'>GET Invoices : </b>". $dataInvoices;  
}
```

11.2 Tiparirea facturilor in format PDF

Metoda **InvoiceDocuments** este de tip GET si returneaza un string care trebuie decodat in base64

Sunt trimise :

InvoiceId - id-ul facturii obtinut din metoda Invoices

Returneaza:

Codul in base64

Exemplu:

```
header("Content-type:application/pdf");
$resultInvoiceDocuments = $urgent->CallMethod('InvoiceDocuments?
InvoiceId=1010219190', $json="", 'GET', $token);
if ($resultInvoiceDocuments['status'] != "200") {
    //echo "<b class='bad'>GET InvoiceDocuments : </b>".
    $resultInvoiceDocuments['message'];
}
else {
    $dataInvoiceDocuments = $resultInvoiceDocuments['message'];
    //echo "<b class='good'>GET InvoiceDocuments : </b>".
    $dataInvoiceDocuments;

    echo base64_decode($dataInvoiceDocuments);
}
```

12 FLUX DE LUCRU PENTRU INTEGRARE

Se apeleaza **PickupLocations** pentru adaugare: Punct ridicare PL1. Id-ul returnat idPL1.

1. Se poate utiliza metoda Localities pentru preluare id localitate

2. Se poate utiliza metoda Streets pentru preluare id Strada

! De observat ca un punct de lucru are AutomaticEOD (inchiderea automata a comenzii) (EOD = end of day)

! Se apeleaza **Awbs** pentru adaugare awb. La sender se utilizeaza idPL1. In acest moment ai o comanda deschisa pe punctul de lucru idPL1.

Comanda se inchide cand:

! se apeleaza **Orders** cu PUT pentru punctul de ridicare idPL1 cu action=1 pentru a lansa comanda.

sau

se atinge AutomaticEOD corespunzator punctului de ridicare

13 ANEXA

SERVICEID

Id	Name
1	Standard
2	Matinal
3	International Rutier
4	Business Partener
5	Import Europa
7	Box 5
8	Dox 0.5
9	Box 10
10	Ship & Go Collect
11	Ship & Go AM
12	Dox 0.5
13	Box 5
14	Box 10
29	Rutier Bulgaria
30	Collect Shop
31	Pre 12

Status: 0 – comanda deschisa

1 – comanda inchisa

OrderType: 1 – online –generat pe Webexpress
2 – telefonica
3 – extern – generat prin API
4- ridicare – generate prin generare” Ridicare de la alta locatie” din
Webexpress
5- email- generate prin email
6 – prestabilita –generata in sistemul intern in fiecare zi