

**Полное описание интеграции с системой DOSTAVKA.guru с помощью WEB-сервисов**

Оглавление

[1. Введение: 3](#_Toc29822246)

[1.1 Получение секретного ключа API 3](#_Toc29822247)

[2. Интеграция в КОРЗИНУ или на страницу оформления заказа интернет магазина 4](#_Toc29822248)

[2.1 Требования к адресу 4](#_Toc29822249)

[2.2 Интерактивная строка ввода адреса с подсказками 4](#_Toc29822250)

[2.3 Вставка карты ПВЗ с событиями для корзины 7](#_Toc29822251)

[2.4 Калькулятор 20](#_Toc29822252)

[2.5 Получение списка ПВЗ по названию города 22](#_Toc29822253)

[2.6 Проверка наличия курьерской доставки в регионы по индексу, городу и названию региона 23](#_Toc29822254)

[3. Интеграция в АДМИН ПАНЕЛЬ интернет магазина 25](#_Toc29822255)

[3.1 Описание полей для отправки данных о заказе 25](#_Toc29822256)

[3.2 Метод отправки данных о заказе 27](#_Toc29822257)

[3.3 Алгоритм получения ответа 28](#_Toc29822258)

[3.4 Распечатка штрих-кодов 30](#_Toc29822259)

[3.5 Проверка статуса созданного заказа в системе 32](#_Toc29822260)

[3.6 Получение ближайших 30 релевантных дат доставки 35](#_Toc29822261)

[3.7 Массовая проверка статусов заказа 36](#_Toc29822262)

[3.8 Печать Акта передачи Заказов курьеру 38](#_Toc29822263)

[3.9 Массовая печать штрих-кодов 39](#_Toc29822264)

[3.10 Отправка разовой заявки на ЗАБОР 40](#_Toc29822265)

[3.11 Получение списка складов с идентификаторами 41](#_Toc29822266)

[4. Дополнительно 43](#_Toc29822267)

[4.1 Получение полного списка ПВЗ и списка пунктов с передачей заказов в курьерскую службу 43](#_Toc29822268)

[4.2 Получение информации о ПВЗ по его ID 46](#_Toc29822269)

[4.3 Получение списка городов с ПВЗ 47](#_Toc29822270)

[4.4 Вставка карты ПВЗ на сайт, как информационная страница 48](#_Toc29822271)

[4.5 Проверка качества адреса 50](#_Toc29822272)

# 1. Введение:

Служба доставка DOSTAVKA.guru предоставляет информационное решение автоматизированного обмена данными между службой доставки и интернет-магазинами. Данное решение позволяет интегрировать рабочую систему интернет магазина с базой DOSTAVKA.guru.

Обмен осуществляется по методу отправки POST данных на сервер и получение соответствующего ответа. Отправка POST данных должны быть в кодировке UTF-8. Ответ от сервера, если нет ошибок, будет передаваться в формате XML, в случае ошибок в ответе сервера будет передаваться стек с ошибками:

*Error:{наименование ошибки 1}*

*Error:{наименование ошибки 2}*

# 1.1 Получение секретного ключа API

Перед применением рабочего ключа API для использования обмена данными, нужно провести интеграцию в тестовом режиме.

Тестовый ID партнера: **9999**

Тестовый ключ-API указан в верхней строке тестового личного кабинета (**Обратите внимание, после получения боевого ключа API, его нужно применить в работе. То есть НЕОБХОДИМО СМЕНИТЬ ТЕСТОВЫЙ КЛЮЧ НА БОЕВОЙ, т.к. тестовый ключ регулярно меняется, что может привести к сбоям в вашей работе**). После успешной интеграции с тестовыми данными, Вы должны подать заявку на получение боевого ключа API менеджерам по электронному адресу info@dostavka.guru с указанием 3-4 номеров заказов, прошедших удачный обмен в тестовом режиме.

Тема письма: Ключ API

Текст письма:

Название интернет-магазина (грузоотпрваитель): Ромашка.рф

ID клиента в системе (логин в личном кабинете): 1234 или cl\_1234

Номера заказов, прошедших обмен: 1234, 5678, а14

# 2. Интеграция в КОРЗИНУ или на страницу оформления заказа интернет магазина

В данном разделе описаны сервисы для автоматизации оформления заказов в системах управления интернет магазинами.

# 2.1 Требования к адресу

Все адреса проходят проверку по базам КЛАДР и ФИАС. Чтобы адрес был правильным и система его пропустила он должен быть:

* Реальным – адрес должен быть зарегистрирован в реестре КЛАДР и ФИАС
* Адрес должен быть полным и указан как минимум до дома
* Адрес не должен содержать: этаж, код домофона, или фразы типа "салон красоты", "сразу за макдональдсом", "территория завода" и пр.

**Схема составления адреса:**

{Регион}, {Район}, {Нас. пункт}, ул. {улица}, д. {дом}, к. {корпус/строение}, кв. {квартира}

**Примеры:**

*Московская обл., Королев, ул. Комитетский Лес, д. 18, к. 1, кв. 247*

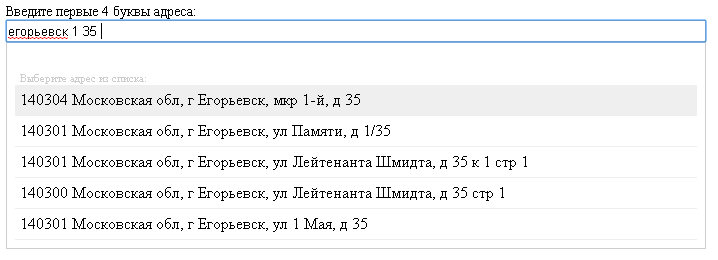
*г Москва, ул Мусы Джалиля, д 2 к 1 кв 3*

# 2.2 Интерактивная строка ввода адреса с подсказками

Автоматизированная строка поиска адреса позволит избежать ошибки при вводе адреса клиентом.

Пример работы скрипта можно посмотреть здесь:

<https://api.dostavka.guru/client/search_addr.html>



**Скрипт вставки поисковой строки:**

Пример html кода со стилями (атрибут id="address2" обязателен для идентификации строки адреса). Блок с классом class="response\_addr": является контейнером для вывода списка подсказок.

<style>

.wrap\_addr{

border:1px solid #cccccc;

padding:8px;

}

.addr\_li:hover{

background-color:#efefef;

}

.cont\_addr\_700, #address2{

width:700px;

}

.cont\_addr\_700 b{

font-size:11px;

color:#cccccc;

margin-left:5px;

font-weight:normal;

}

</style>

<div class="cont\_addr\_700">

<div style="font-size:14px;">Введите первые 4 буквы адреса:</div>

<input autocomplete="off" name="adres" id="address2" type="text" value="">

<div class="response\_addr">

</div>

</div>

Скрипт для получения списка подсказок и извлечения всех параметров адреса(Для работы скрипта необходимо наличие подключенной библиотеки jquery с поодержкой конструкции "*$(document).on('...*"). В функции get\_result\_addr\_list\_GURU() - необходимо поставить Ваши ключ API и ID партнера:

<script>

//-- Обработка ответа

function onAjaxSuccessAddrList(data){

$('.response\_addr').html('<div class="wrap\_addr">'+data+'</div>');

}

//-- ----------------

//-- Ajax запрос с выводом прелоадера

function get\_result\_addr\_list\_GURU(focus\_value){

var key='cc03e747a6afbbcbf8be7668acfebee5';//Ваш ключ

var id='9999';//Ваш id

$('.response\_addr').html('<div style="padding:18px;"><img src="https://api.dostavka.guru/client/712m.GIF"></div>');

$.post("https://api.dostavka.guru/client/dd\_2\_ch.php", { str: focus\_value, key: key, id: id }, onAjaxSuccessAddrList);

}

//-- ---------------------------------

$(document).ready(function(){

//-- Отлавливаем нажатие клавиш на поле адреса

$(document).keyup(function(event){

var focus\_name=$("input:focus").attr('id');

var focus\_value=$("input:focus").val();

if(focus\_name=='address2'){

if(focus\_value.length>3){

get\_result\_addr\_list\_GURU(focus\_value);

}

}

});

//-- -----------------------------------------

//-- Получение параметров адреса после клика по выпадающщему списку

$(document).on('click', '.addr\_li', function() {

var region\_v=$(this).attr('param\_reg');//Регион

var raion\_v=$(this).attr('param\_rai');//Район

var city\_v=$(this).attr('param\_cit');//Населенный пункт

var zip\_v=$(this).attr('param\_zip');//Индекс

var korpus=$(this).attr('param\_kor');//Корпус дома

var flat=$(this).attr('param\_fla');//Квартира

var street=$(this).attr('param\_stre');//улица

var house=$(this).attr('param\_hou');//Дом

var kladr=$(this).attr('param\_kladr');//Идентификатор КЛАДРа

var type\_city=$(this).attr('param\_type');//Тип населенного пункта

var adres=$(this).attr('param\_adres');//Распознанная часть адреса

$('#address2').val(adres);

if(house){

//alert('Вы выбрали: ['+zip\_v+'] '+adres);

}else{

//alert('Введите/выберите адрес с точностью до дома');

}

$('.response\_addr').html('');

});

//-- -------------------------------------------------

});

</script>

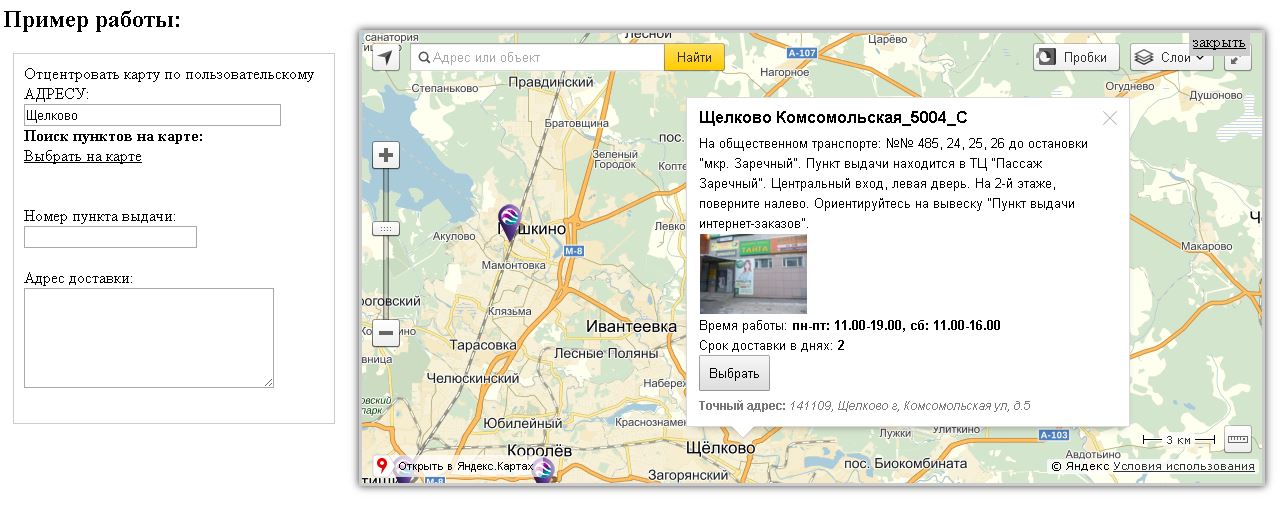
# 2.3 Вставка карты ПВЗ с событиями для корзины

**Вариант 1 (стандартные сроки доставки):**

Данный пример позволит вставить карту ПВЗ в страницу оформления заказа и перехватить нужные параметры выбранного пункта.

Пример работы скрипта и более удобное описание можно посмотреть здесь:

<https://api.dostavka.guru/client/map_with_but/map_with_but_2.html>



Карта и кнопки открывающие и закрывающие карту(Вставляется внутри тега "body" в нужном для Вас месте):

<b>Поиск пунктов на карте:</b>

<br><span id="close\_map" style="position:fixed; top:-2000px; cursor:pointer; z-index:999; right:75px; background:#cccccc; display:inline-block; padding:2px 4px; padding-bottom:4px; text-decoration:underline;">закрыть</span>

<span style="cursor:pointer; display:block; text-decoration:underline;" class="message-map-link">Выбрать на карте</span>

<div id="YMapsID"></div>

Первоначальный стиль скрывающий карту(Вставляется внутри тега "head"):

<style>

#YMapsID{

z-index:998;

display:block;

position:fixed;

top:-2000px;

width: 900px;

height: 450px;

margin:10px;

border:solid 3px #cccccc;

right:50px;

box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

-moz-box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

-webkit-box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

}

</style>

Скрипт инициализации карты(первая строчка не нужна если jQuery уже подключен)(Вставляется внутри тега "head" на нужной для Вас странице). Для того чтобы карта центрировалась по гео объекту, выбранному пользователем, нужно применить элементу формы с адресом или городом класс"**geo\_class**":

<script src="https://api.dostavka.guru/jquery.js"></script>

<script src="https://api.dostavka.guru/client/collection-search-provider.js"></script>

<script src="https://api-maps.yandex.ru/2.1/?load=package.standard,package.geoObjects&lang=ru-RU" type="text/javascript"></script>

<script type="text/javascript">

//Создание точек на карте------

function maps\_init\_GURU(points, center\_1, center\_2){

if(center\_1==''){

var center\_1=55.755768;

var center\_2=37.617671;

}

ymaps.ready(init);

function init() {

var myMap = new ymaps.Map('YMapsID', {

center: [center\_1, center\_2],

zoom: 8,

behaviors: ["default", "scrollZoom"]

}),

collection = new ymaps.GeoObjectCollection();

myMap.controls.add(

new ymaps.control.ZoomControl()

);

myMap.geoObjects.add(collection);

for(var i = 0, len = points.length; i < len; i++) {

collection.add(

new ymaps.Placemark(points[i].coords, {

balloonContentHeader: points[i].label+' ',

balloonContentBody: ''+points[i].way\_desc+'<div>Время работы: <b>'+points[i].time+'</b><br>'+points[i].params+'Срок доставки в днях: <b>'+points[i].days+'</b><br><input style="padding:8px;" type="button" pf="'+points[i].pf+'" value="Выбрать" class="select-point" rel="'+points[i].id+'" city="'+points[i].city+'" name="'+points[i].label+'" region="'+points[i].region+'" date="'+points[i].date+'"> </div>',

balloonContentFooter: '<b>Точный адрес:</b> <i>'+points[i].desc+'</i>',

hintContent: points[i].label,

searchStr: '<b>'+points[i].label+'</b> '+points[i].desc+'<br>'

},

{

iconLayout: 'default#image',

iconImageHref: 'https://dostavka.guru/map\_icon2.png',

iconImageSize: [31, 40],

iconImageOffset: [-20, -20],

// Определим интерактивную область над картинкой.

iconShape: {

type: 'Circle',

coordinates: [0, 0],

radius: 20

}

}

)

);

}

document.getElementById('message-map-link').onclick = function () {

if($('.geo\_class').val()){

var myGeocoder = ymaps.geocode($('.geo\_class').val()+' Россия');

var ccc=myGeocoder.then(

function (res) {

if(res.geoObjects.get(0).geometry.getCoordinates()!=''){

myMap.setCenter(res.geoObjects.get(0).geometry.getCoordinates(), 11, {

checkZoomRange: true

});

}else{}

},

function (err) {

alert('Ошибка: объект на карте не найден!');

}

);

}else{}

};

}

}

//---------------------------------------

//ОТКРЫТЬ КАРТУ

function open\_GURU\_map(){

$('#YMapsID').css('top', '100px');

$('#close\_map').css('top', '110px');

return false;

}

//------------

//ЗАКРЫТЬ КАРТУ

function close\_GURU\_map(){

$('#YMapsID').css('top', '-2000px');

$('#close\_map').css('top', '-2000px');

return false;

}

//------------

//создание карты

function new\_map\_new\_center(){

$.post("https://api.dostavka.guru/client/get\_pvz\_codes\_2.php",

{init: 'get\_pvz' }).success(function(data) {

var center\_1='37.617671';

var center\_2='55.755768';

var points = eval("obj = " + data);

if(data==''){

alert('Нет соединения с сервером пунктов выдачи!');

return false;

}

maps\_init\_GURU(points, center\_1, center\_2);

open\_GURU\_map();

});

}

$(document).ready(function(){

$.post("https://api.dostavka.guru/client/get\_pvz\_codes\_2.php",

{init: 'get\_pvz' }).success(function(data) {

var center\_1='';

var center\_2='';

var points = eval("obj = " + data);

if(data==''){

alert('Нет соединения с сервером пунктов выдачи!');

return false;

}

maps\_init\_GURU(points, center\_1, center\_2);

});

});

//--------------

$(document).ready(function(){

//ТРИГЕРЫ

$('.message-map-link').live('click', function(){

open\_GURU\_map();//открыть каррту

return false;

});

$("#close\_map").live('click', function(){

close\_GURU\_map();//закрыть карту

return false;

});

//ПОЛУЧИТЬ ДАННЫЕ ПО ВЫБОРУ ПУНКТА

$('.select-point').live('click', function(){

var code=$(this).attr('rel');//Код пвз

var city=$(this).attr('city');//Город пвз

var name=$(this).attr('name');//Наименование пвз

var region=$(this).attr('region');//Регтон пвз

var date\_pvz=$(this).attr('date');//Ближайшая дата доставки

//Здесь код, который заполнит нужные поля Вашей информационной системы

alert('Выбран пункт: '+code);

//-------------------------------------------------------------------

close\_GURU\_map();//закрыть карту

return false;

});

});

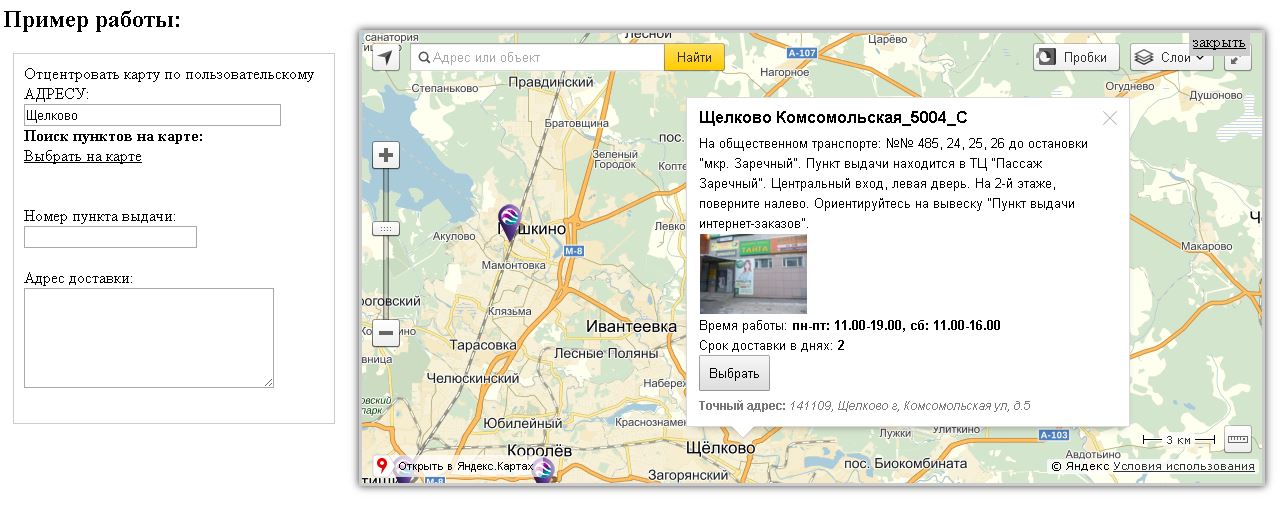
</script>

**Вариант 2 (пользовательские сроки доставки):**

Данный пример позволит вставить карту ПВЗ в страницу оформления заказа и перехватить нужные параметры выбранного пункта. В ДВУХ ФУНКЦИЯХ КОДА НУЖНО ВСТАВИТЬ ВАШ **ID ПАТНЕРА** В СИСТЕМА DOSTAVKA.GURU. После чего к сроку доставки будет плюсоваться заданный Вами интервал в личном кабинете в “Настройка тарифов ” -> “Увеличение срока доставки в днях для карты ПВЗ” .

Пример работы скрипта и более удобное описание можно посмотреть здесь:

<https://api.dostavka.guru/client/map_with_but/map_with_but_4.html>



Карта и кнопки открывающие и закрывающие карту(Вставляется внутри тега "body" в нужном для Вас месте):

<b>Поиск пунктов на карте:</b>

<br><span id="close\_map" style="position:fixed; top:-2000px; cursor:pointer; z-index:999; right:75px; background:#cccccc; display:inline-block; padding:2px 4px; padding-bottom:4px; text-decoration:underline;">закрыть</span>

<span style="cursor:pointer; display:block; text-decoration:underline;" class="message-map-link">Выбрать на карте</span>

<div id="YMapsID"></div>

Первоначальный стиль скрывающий карту(Вставляется внутри тега "head"):

<style>

#YMapsID{

z-index:998;

display:block;

position:fixed;

top:-2000px;

width: 900px;

height: 450px;

margin:10px;

border:solid 3px #cccccc;

right:50px;

box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

-moz-box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

-webkit-box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

}

</style>

Скрипт инициализации карты(первая строчка не нужна если jQuery уже подключен)(Вставляется внутри тега "head" на нужной для Вас странице). Для того чтобы карта центрировалась по гео объекту, выбранному пользователем, нужно применить элементу формы с адресом или городом класс"**geo\_class**":

<script src="https://api.dostavka.guru/jquery.js"></script>

<script src="https://api.dostavka.guru/client/collection-search-provider.js"></script>

<script src="https://api-maps.yandex.ru/2.1/?load=package.standard,package.geoObjects&lang=ru-RU" type="text/javascript"></script>

<script type="text/javascript">

//Создание точек на карте------

function maps\_init\_GURU(points, center\_1, center\_2){

if(center\_1==''){

var center\_1=55.755768;

var center\_2=37.617671;

}

ymaps.ready(init);

function init() {

var myMap = new ymaps.Map('YMapsID', {

center: [center\_1, center\_2],

zoom: 8,

behaviors: ["default", "scrollZoom"]

}),

collection = new ymaps.GeoObjectCollection();

myMap.controls.add(

new ymaps.control.ZoomControl()

);

myMap.geoObjects.add(collection);

for(var i = 0, len = points.length; i < len; i++) {

collection.add(

new ymaps.Placemark(points[i].coords, {

balloonContentHeader: points[i].label+' ',

balloonContentBody: ''+points[i].way\_desc+'<div>Время работы: <b>'+points[i].time+'</b><br>'+points[i].params+'Срок доставки в днях: <b>'+points[i].days+'</b><br><input style="padding:8px;" type="button" pf="'+points[i].pf+'" value="Выбрать" class="select-point" rel="'+points[i].id+'" city="'+points[i].city+'" name="'+points[i].label+'" region="'+points[i].region+'" date="'+points[i].date+'"> </div>',

balloonContentFooter: '<b>Точный адрес:</b> <i>'+points[i].desc+'</i>',

hintContent: points[i].label,

searchStr: '<b>'+points[i].label+'</b> '+points[i].desc+'<br>'

},

{

iconLayout: 'default#image',

iconImageHref: 'https://dostavka.guru/map\_icon2.png',

iconImageSize: [31, 40],

iconImageOffset: [-20, -20],

// Определим интерактивную область над картинкой.

iconShape: {

type: 'Circle',

coordinates: [0, 0],

radius: 20

}

}

)

);

}

document.getElementById('message-map-link').onclick = function () {

if($('.geo\_class').val()){

var myGeocoder = ymaps.geocode($('.geo\_class').val()+' Россия');

var ccc=myGeocoder.then(

function (res) {

if(res.geoObjects.get(0).geometry.getCoordinates()!=''){

myMap.setCenter(res.geoObjects.get(0).geometry.getCoordinates(), 11, {

checkZoomRange: true

});

}else{}

},

function (err) {

alert('Ошибка: объект на карте не найден!');

}

);

}else{}

};

}

}

//---------------------------------------

//ОТКРЫТЬ КАРТУ

function open\_GURU\_map(){

$('#YMapsID').css('top', '100px');

$('#close\_map').css('top', '110px');

return false;

}

//------------

//ЗАКРЫТЬ КАРТУ

function close\_GURU\_map(){

$('#YMapsID').css('top', '-2000px');

$('#close\_map').css('top', '-2000px');

return false;

}

//------------

//создание карты

function new\_map\_new\_center(){

$.post("https://api.dostavka.guru/client/get\_pvz\_codes\_3.php",

{init: 'get\_pvz', id: '**9999**' }).success(function(data) {

var center\_1='37.617671';

var center\_2='55.755768';

var points = eval("obj = " + data);

if(data==''){

alert('Нет соединения с сервером пунктов выдачи!');

return false;

}

maps\_init\_GURU(points, center\_1, center\_2);

open\_GURU\_map();

});

}

$(document).ready(function(){

$.post("https://api.dostavka.guru/client/get\_pvz\_codes\_3.php",

{init: 'get\_pvz', id: '**9999**' }).success(function(data) {

var center\_1='';

var center\_2='';

var points = eval("obj = " + data);

if(data==''){

alert('Нет соединения с сервером пунктов выдачи!');

return false;

}

maps\_init\_GURU(points, center\_1, center\_2);

});

});

//--------------

$(document).ready(function(){

//ТРИГЕРЫ

$('.message-map-link').live('click', function(){

open\_GURU\_map();//открыть каррту

return false;

});

$("#close\_map").live('click', function(){

close\_GURU\_map();//закрыть карту

return false;

});

//ПОЛУЧИТЬ ДАННЫЕ ПО ВЫБОРУ ПУНКТА

$('.select-point').live('click', function(){

var code=$(this).attr('rel');//Код пвз

var city=$(this).attr('city');//Город пвз

var name=$(this).attr('name');//Наименование пвз

var region=$(this).attr('region');//Регтон пвз

var date\_pvz=$(this).attr('date');//Ближайшая дата доставки

//Здесь код, который заполнит нужные поля Вашей информационной системы

alert('Выбран пункт: '+code);

//-------------------------------------------------------------------

close\_GURU\_map();//закрыть карту

return false;

});

});

</script>

# 2.4 Калькулятор

**Передаваемые параметры:**

Все передаваемые параметры должны иметь кодировку utf-8 и передаваться POST методом на скрипт: <https://api.dostavka.guru/client/calc_guru_main_3_0.php>

Для передачи данных в формате JSON:

https://api.dostavka.guru/client/calc\_guru\_main\_3\_0.php?json=yes

Передаваемые параметры:

'client' – Ваш ID партнера в системе DOSTAVKA.guru

'key' – Ключ доступа

'method' - Обязательное поле – способ доставки. Может выбираться из списка: Курьер, Почта, ПВЗ

'weight' - Обязательное поле – вес заказа в кг в формате “0.00”.

'point' - Номер пункты ПВЗ. Поле обязательно для доставки до ПВЗ

'ocen\_sum' - Оценочная стоимость в формате “0.00”

'nal\_plat' – Наложенный платеж в формате “0.00”

'zip\_1' – Индекс отправителя

'adres' – Полная строка адреса

'night' – Индикатор ночной доставки (0 или пустое значение – ночная доставка не требуется, 1 – ночная доставка)

'post\_type\_otp' – тип отправления только для посылок ("1"-Нестандартная\*, "2"- Нестандартная\*\*, негабаритная, "3"- 1 класс\*\*\*)

**\* - Нестандартная** - обычная посылка

**\*\* -** **Нестандартная, негабаритная** - сумма трех сторон отправления превышает 120 см или одна из сторон превышает 60 см или вес превышает 10 кг

\*\*\***-1 класс** - Предельная масса - 2.5 кг. Минимальный размер - 110 х 190 мм. Максимальный размер - не более 36 см в одном из трех измерений, сумма трех измерений не должна превышать 70 см. Сумма объявленной ценности для посылки 1-го класса не должна превышать 100000 руб. Сумма наложенного платежа не должна превышать сумму объявленной ценности почтового отправления.

**Обработка ответа:**

Ответ передается одной строкой с разделителем “::”

ЦЕНА РУБ::СРОК ДОСТАВКИ::ПРИЕМ ТОЛЬКО ОПЛАЧЕННЫХ ЗАКАЗОВ::ВОЗМОЖНОСТЬ ОПЛТЫ КАРТОЙ

Пример ответа:

215::5 дн.::only\_paid=0::acquiring=0

Где - стоимость доставки: **215 руб.**, срок доставки **5 дн.**, Только предоплаченные заказы: **НЕТ**, Есть ли прием банковских карт: **НЕТ**

Параметры ***ПРИЕМ ТОЛЬКО ОПЛАЧЕННЫХ ЗАКАЗОВ*** и ***ВОЗМОЖНОСТЬ ОПЛТЫ КАРТОЙ***– актуальны только для доставки заказов курьером в регионы и доставки заказов в пункты выдачи. Значения этих полей 0 или 1 – НЕТ, ДА соответственно

При возникновении ошибки ответ буде выглядеть так: ERROR::**Не задан индекс получателя!**

Где первое значение ERROR – является идентификатором ошибки.

**Пример расчета Курьерской доставки по Москве и МО, Санкт-Петербургу и ЛО (все поля являются обязательными):**

**Отправляемые данные:**

$\_POST=array(

'method'=>'Курьер',

'weight'=>'1',

'ocen\_sum'=>'1000',

'nal\_plat'=>'1000',

'client'=>'9999',

'key'=>'cc03e747a6afbbcbf8be7668acfebee5',

'adres'=>'140400 Московская обл, г Коломна, снт Волна-Коломна, д. 1'

);

Ответ: 715::1 дн.

**Пример расчета Курьерской доставки в регионы (все поля являются обязательными):**

**Отправляемые данные:**

$\_POST=array(

'method'=>'Курьер',

'weight'=>'1',

'ocen\_sum'=>'1000',

'nal\_plat'=>'1000',

'client'=>'9999',

'key'=>'cc03e747a6afbbcbf8be7668acfebee5',

'adres'=>'г. Новосибирск, Улица Курганская, дом 38',

);

Ответ: 475::7 дн.::only\_paid=0::acquiring=0

**Пример расчета доставки до ПВЗ (все поля являются обязательными):**

**Отправляемые данные:**

$\_POST=array(

'method'=>'ПВЗ',

'weight'=>'1',

'ocen\_sum'=>'1000',

'nal\_plat'=>'1000',

'client'=>'9999',

'key'=>'cc03e747a6afbbcbf8be7668acfebee5',

'point'=>'30021'

);

Ответ: 215::5 дн.::only\_paid=0::acquiring=0

**Пример расчета доставки почтой (все поля являются обязательными):**

**Отправляемые данные:**

$\_POST=array(

'method'=>'Почта',

'weight'=>'5',

'ocen\_sum'=>'1000',

'nal\_plat'=>'1000',

'client'=>'9999',

'key'=>'cc03e747a6afbbcbf8be7668acfebee5',

'zip\_1'=>'101000',

'adres'=>'г Новосибирск, ул Магаданская, д 1',

'post\_type\_otp'=>'1'

);

Ответ: 626::10 дн.

# 2.5 Получение списка ПВЗ по названию города

Получение списка ПВЗ по наименованию города по принципу POST запроса, с указанием обязательных переменных.

**URL для отправки POST массива:**

<https://api.dostavka.guru/client/pvz_list.php>

Для передачи данных в формате JSON:

<https://api.dostavka.guru/client/pvz_list.php>?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *script* | тип скрипта (**all\_list** - список ПВЗ) | **Текстовый** |
| *city\_name* | Город ПВЗ | **Текстовый** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[script] => all\_list

[city\_name] => Калуга

)

**Пример ответа сервера:**

Ответ сервера такой же, как в пункте **3.3**

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
и прочие...

# 2.6 Проверка наличия курьерской доставки в регионы по индексу, городу и названию региона

Проверка на наличие курьерской доставки по принципу POST запроса, с указанием обязательных переменных.

**URL для отправки POST массива:**

<https://api.dostavka.guru/client/get_kd_API_guru.php>

Для передачи данных в формате JSON:

<https://api.dostavka.guru/client/get_kd_API_guru.php>?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *region* | Регион | **Текстовый UTF-8** |
| *city* | Город | **Текстовый UTF-8** |
| *zip* | Индекс | **6-ти значный почтовый индекс** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[region] => Челябинская обл

[city] => Челябинск

[zip] => 454000

)

**Пример ответа сервера:**

“0” – курьерской доставки нет

“1”, “2”, “10” – если больше нуля - курьерская доставка есть, ответ сервера - срок доставки в днях

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
и прочие...

# 3. Интеграция в АДМИН ПАНЕЛЬ интернет магазина

Данные сервисы помогут Вам создать автоматизированную систему отправки заказа на доставку прямо из администраторской интернет магазина.

# 3.1 Описание полей для отправки данных о заказе

Таблица полей передаваемых данных:

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. Выдается по запросу на почту info@dostavka.guru. | **Хеш код** |
| *order\_number* | Номер заказа в Вашей информационной системе. Обязательно должен быть индивидуальным для каждого заказа, кроме услуги ЗАБОР | **Текстовый (максимум 30 символов)** |
| *usluga* | Услуга. Выбирается из списка | На выбор: **ЗАБОР, ДОСТАВКА, ВОЗВРАТ, Ночная доставка, Экспресс доставка, Экспресс возврат** |
| *sposob\_dostavki* | Способ доставки. Выбирается из списка. | На выбор: **Курьер, Почта, ПВЗ** |
| *cont\_name* | ФИО клиента получающего заказ. | **Текстовый. Если способ доставки “Почта”, то обязательно требуется указание отчества** |
| *cont\_mail* | E-mail клиента | **Текстовый. Общепринятый формат электронной почты:**  ***name@*** ***domain.com*** |
| *kgt* | Наличие Крупно Габаритного Товара | **Булевый: "0" (не нужен), "1" (нужен)** |
| *nds* | НДС (non, zero, 10, 18, 20) | **Выбирается из списка:**  **non=>БЕЗ НДС,**  **zero=>0%,**  **10=>10%,**  **20=>20%** |
| *tip\_otpr* | Тип отправления (обязателен если выбрана Почта) | На выбор: **разное, 1 класс** |
| *cont\_tel* | Контактный телефон клиента получающего заказ. Для указания второго номера телефона используйте "," или ";" | **формат: 8 (999) 9999999,+7 (999) 999-99-99,+7 (999) 999 99 99** |
| *ves\_kg* | Вес в килограммах (кг). При доставке курьером в регионы и при способе доставки ПВЗ вес не должен быть больше 15 кг | **Числовой до сотой: 0.00** |
| *guid* | Индивидуальный идентификатор в Вашей системе | **Строковый, не обязательный** |
| *lifting* | Подъем – актуально для заказов более 25 кг | **Булевый: "0" (не нужен), "1" (нужен)** |
| *mesta* | Количество мест. Не может быть меньше или равен нулю. При способе доставки "**Почта**" - количество мест должно быть равным 1 | **Числовой целый: 1** |
| *date\_dost* | Дата доставки. | **YYYY.MM.DD** |
| *delivery\_time\_from* | Время доставки С. (При доставке заказа курьером в регионы интервал времени доставки возможен только с 10:00 до 14:00 и с 14:00 до 18:00. При ночной доставке возможны следующие интервалы: с 22:00 до 02:00 и с 02:00 до 06:00. При экспресс доставке возможны следующие интервалы: с 16:00 до 19:00 и с 19:00 до 22:00). Подробнее об интервалах [здесь](http://dostavka.guru/docs/pravila_osush_dostavki_19_04_2019.pdf?677947). | **hh:mm** |
| *delivery\_time\_to* | Время доставки ПО. (При доставке заказа курьером в регионы интервал времени доставки возможен только с 10:00 до 14:00 и с 14:00 до 18:00. При ночной доставке возможны следующие интервалы: с 22:00 до 02:00 и с 02:00 до 06:00 При экспресс доставке возможны следующие интервалы: с 16:00 до 19:00 и с 19:00 до 22:00). Подробнее об интервалах [здесь](http://dostavka.guru/docs/pravila_osush_dostavki_19_04_2019.pdf?677947). | **hh:mm** |
| *region\_iz* | Регион ИЗ. Регион откуда повезется заказ. | Выбирается из списка: **Москва, Санкт-Петербург** |
| *int\_pvz* | code ПВЗ с возможностью передачи заказов в курьерскую службу | **Текстовый** |
| *zip\_v* | Индекс. Если способом доставки выбрана почта или курьер, то поле индекс обязательно для заполнения. | **Текстовый** |
| *punkt\_vivoz* | Пункт самовывоза (только для отправлений на ПВЗ) | **Числовой. ID пункта.** |
| *comment* | Примечание или комментарий к доставке. | **Текстовый – максимум 200 символов** |
| *adres* | Точный адрес доставки. | **Текстовый** Примеры: ***г Москва, ул Мусы Джалиля, д 2 к 1 кв 3*** или  ***Московская обл, г Егорьевск, мкр 1-й, д 35*** или  ***Московская обл, Серпуховский р-н, деревня Большая Городня, ул Дачная, д 1*** |
| *code\_{N}* | Штрих кода. Для каждого места задается отдельно, где {N} - номер места. Не обязательный параметр. В случае передачи пустых полей штрих-коды генерируются автоматически и выводятся в ответе сервера. | **Штрих код в формате EAN 13** |
| *nal\_plat* | Наложенный платеж. | **Числовой до сотой: 0.00** |
| *ocen\_sum* | Оценочная стоимость. | **Числовой до сотой: 0.00** |
| *on\_error* | Стоимость ошибочной доставки - стоимость доставки для клиента при полном отказе от заказа в момент приезда курьера на адрес (кроме причин: "Отсутствие клиента", "Нет денег", "Девиантное поведение", "Некорректный адрес"). | **Числовой до сотой: 0.00** |
| *prepay* | Предоплата заказа | **Числовой до сотой: 0.00** |
| *docs\_return* | Требуется возврат документов (передается, только если требуется возврат документов) | **Если требуется возврат документов, то значение “Y”** |
| *warehouse\_id* | Идентификатор склада возврата если требуется возврат на неосновной склад (см. пункт 3.10) | **Строковый. Пример:**  123243a-3a57-11e7-950f-000c4754834 |
| *dop\_limit* | Минимальная сумма выкупа (МСВ), при которой доставка для клиента будет бесплатной. | **Числовой до сотой: 0.00** |
| *dop\_tarif* | Стоимость доставки для клиента при оплате МСВ. Стоимость доставки не входит в расчет МСВ. | **Числовой до сотой: 0.00** |
| **Поля для потоварного учета** | | |
| *name\_good\_{N}* | Наименование товара (для потоварного учета) | **Текстовый (обязательный)** |
| *qty\_good\_{N}* | Количество товара (для потоварного учета) | **Целое число (обязательный)** |
| *ed\_good\_{N}* | Единица измерения (для потоварного учета) | **шт., уп., кг., коробка, м, м куб.**  **(обязательный)** |
| *code\_good\_{N}* | Артикул товара (для потоварного учета) | **Текстовый (обязательный)** |
| *price\_good\_{N}* | Цена за единицу товара (для потоварного учета) | **Числовой до сотой: 0.00 (обязательный)** |
| *bare\_good\_{N}* | Штрих-код позиции (для потоварного учета) | **Текстовый, разделитель ";" (точка с запятой)** |
| *har\_good\_{N}* | Характеристики товара (для потоварного учета) | **Текстовый** |
| *ndsch\_good\_{N}* | НДС (0, 10, 20) (для потоварного учета) | **Выбирается из списка:**  **0=>0%,**  **10=>10%,**  **20=>20%** |
| *ch\_dostavka* | Стоимость доставки (для потоварного учета) | **Целое число**  **(может быть 0)** |
| **{N} –** идентификатор или порядковый номер товарной единицы | | |

# 3.2 Метод отправки данных о заказе

Отправка данных осуществляется методом отправки POST массива на сервер. Параметры отправки данных:

Адрес приема: [**https://api.dostavka.guru/client/in\_up\_2\_0.php**](https://api.dostavka.guru/client/in_up_2_0.php)

Для передачи данных в формате JSON:

[**https://api.dostavka.guru/client/in\_up\_2\_0.php**](https://api.dostavka.guru/client/in_up_2_0.php)?json=yes

Пример POST массива переменных:

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[usluga] => ДОСТАВКА

[kgt] => 0

[order\_number] => no-0000025

[warehouse\_id] => 123243a-3a57-11e7-950f-000c4754834

[sposob\_dostavki] => Курьер

[docs\_return] => Y

[date\_dost] => 2015.02.18

[delivery\_time\_from] => 10:00

[delivery\_time\_to] => 14:00

[region\_iz] => Москва

[zip\_v] => 140304

[adres] => Московская обл., г. Егорьевск, 1 микрорайон, д. 35, кв. 34

[punkt\_vivoz] =>

[cont\_name] => Иванов Иван

[cont\_tel] => +7 (985) 134 03 64

[mesta] => 3

[code\_1] => 4444444444444

[code\_2] => 2222222222222

[code\_3] => 3333333333333

[ves\_kg] => 5

[nal\_plat] => 1346

[ocen\_sum] => 1346

[ch\_dostavka] => 550

[name\_good\_1] => Чашка белая с ручкой

[qty\_good\_1] => 1

[ed\_good\_1] => шт.

[code\_good\_1] => art.1245

[price\_good\_1] => 256

[har\_good\_1] => Черная ручка, синяя каёмка

[name\_good\_2] => Чашка белая с ручкой

[qty\_good\_2] => 2

[ed\_good\_2] => шт.

[code\_good\_2] => art.1246

[price\_good\_2] => 270

[har\_good\_1] => Красная ручка, золотая каёмка

)

# 3.3 Алгоритм получения ответа

Пример ответа сервера при удачной загрузке заказа:

<?xml version="1.0" encoding="UTF-8"?>  
<root>  
<order>  
<order\_id>no-0000025</order\_id>

<track\_id>00A45FB8</track\_id>  
<uid>uid\_415e35a0-b292-11e4-8e8f-386077e5d2cc</uid>

<doc>ДОК(qwe123eeeeeew12):::9999000644110</doc>  
<location\_status>У курьера</location\_status>  
<order\_status>Перенос</order\_status>

<track\_number>RN1234534534234CN</track\_number>  
<barcodes>  
<code>9999000005515</code>  
<code>9999000005522</code>  
<code>9999000005539</code>  
</barcodes>

<payment\_delivery>100</payment\_delivery>

<goods>

<product>

<name>Футболка серая</name>

<code>47ааа</code>

<characteristic>черная</characteristic>

<amount>2</amount>

<unit>шт</unit>

<price>200</price>

<desc>Майка порвана</desc>

<return>0</return>

</product>

<product>

<name>Носки прочные</name>

<code>33577</code>

<characteristic>лайкра</characteristic>

<amount>3</amount>

<unit>шт</unit>

<price>500</price>

<desc>Носки грязные</desc>

<return>0</return>

</product>

</goods>  
</order>  
</root>

Где "order\_id" - номер заказа в Вашей информационной системе, "uid" - дополнительный идентификатор в системе dostavka.guru, "location\_status" - местоположение, "order\_status" - статус заказа, "doc" – номер заказа и штрих код заявки на возврат документов, "barcodes" - массив штрих кодов, "track\_number " – идентификатор отправления, "payment\_delivery" – стоимость доставки при частичной выдаче, "track\_id" – идентификатор трека для отслеживания посылки, "goods" – блок с позициями в частичной выдаче который включает в себя блоки "product" с параметрами:

"name" – наименование товара

"code" – артикул товара

"characteristic" – характеристики товара

"amount" – количество товаров

"unit" – единица

"price" – цена за единицу

"desc" – описание результата доставки от курьера

"return" – количество возврата

Список возможных статусов и местоположений:

Статусы заказа:  
**Пустой** - Заказ в обработке  
**Перенос** - Заказ перенесен на другую дату  
**Отказ** - Клиент отказался от Заказа до момента доставки курьером по адресу  
**Частично доставлен** - Заказ доставлен клиенту, но клиент выкупил только часть заказа  
**Ошибочно доставлен** - Клиент отказался от доставки в момент приезда курьера или не вышел на связь при приезде курьера.  
**Доставлен** - Заказ доставлен, вручен Клиенту или передан Партнёру (транспортные компании, Почта РФ)  
Статусы местоположений:  
**Ожидание** - Заказ загружен в систему, ожидается поступление Заказа на склад  
**Непоступил** - Заказ должен был поступить на наш склад на указанную дату доставки, но не пришел на наш склад  
**На складе** - Заказ находится на нашем складе  
**В пути** - Заказ отправлен в распределительный центр в городе отправки

**На складе ОП** – Заказ находится на распределительном центре в городе отправки

**Магистраль** – Заказ движется от/к распределительного центра в город доставки

**На складе филиала** – Заказ поступил на склад в городе доставки и ожидает отправку до адресата

**На ПВЗ -** Заказ находится в Пункте Выдачи Заказов

**У курьера** - Заказ находится у курьера для осуществления доставки

**У клиента** - Клиент получил Заказ

**Возврат в пути** - возвратный Заказ отправлен из региона доставки или ожидается приход такого Заказа от партнера  
**Возврат у курьера**- возвратный Заказ у курьера для передачи в ИМ

**Возвращен в ИМ** - Возвратный Заказ передан в интернет-магазин

**Отказ ИМ** – интернет-магазин отказался от получения возврата по причине недостачи или порчи.

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
Error: 300 - Услуга указана не верно!  
и прочие...

Отследить ошибку можно просто разбив строку ответа на массив по двоеточию ":". Если Первым элементом массива будет строка "Error", значит сервер вернул сообщение об ошибке.

# 3.4 Распечатка штрих-кодов

Вы можете распечатать штрих код не прибегая к использованию личного кабинета! Для этого нужно выполнить GET запрос к системе API.

**! На данном этапе реализована печать только штрих кодов формата EAN 13.**

**URL для вызова страницы с штрих кодом:**

<https://api.dostavka.guru/client/barcode/barcode.php>

**GET переменные:**

**barcode** - штрих код  
**orderID** - номер заказа в Вашей информационной системе  
**clID** - Ваш partner id в системе dostavka.guru  
**data** - дата доставки в формате YYYY-MM-DD  
**sposob** - способ доставки (Курьер, Почта, ПВЗ)  
**mesta** - количество мест  
**mesto** - номер места

**Пример строки запроса:**

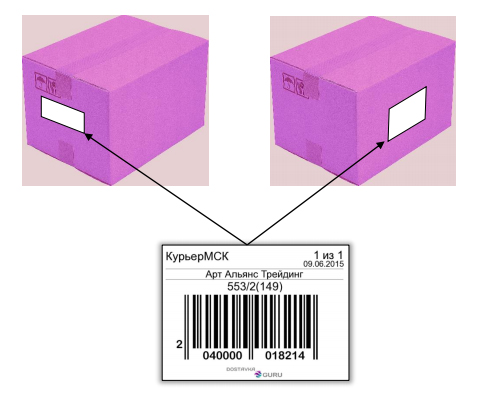
[http://api.dostavka.guru/client/barcode/barcode.php?barcode=9999000005225&mesto=2&orderID=testAPI-in5-FULL&clID=9999&data=2015-02-13&sposob=Курьер&mesta=3](http://api.dostavka.guru/client/barcode/barcode.php?barcode=9999000005225&mesto=2&orderID=testAPI-in5-FULL&clID=9999&data=2015-02-13&sposob=%D0%9A%D1%83%D1%80%D1%8C%D0%B5%D1%80%D0%9C%D0%A1%D0%9A&mesta=3)

**Ответ:**



**! В случае передачи не корректных параметров этикетка не сгенерируется.**

Вы можете использовать собственные методы печати штрих-кодов по заданному описанию:



Каждое отправление должно быть промаркировано (нанесен ярлык). Заказ может состоять

из нескольких мест. Каждое место должно быть промаркировано ярлыками. Маркировка (ярлык)

должен содержать следующую обязательную информацию:

- номер заказа

- штрих-код заказа

- номер места и общее количество мест в заказе (если заказ состоит из одного места,

указывается «1 из 1». Если заказ состоит двух мест, на ярлыках указывается «1 из 2» и «2 из 2»)

- наименование отправителя заказа (Наименование грузоотправителя в системе

DOSTAVKA.GURU)

- Дата доставки

- Способ доставки

**Правила написания штрих-кода:**

1. Штрих-код должен быть в формате EAN-13 <https://ru.wikipedia.org/wiki/European_Article_Number>
2. Штрих-код содержит только цифры и состоит из 13 символов

**Схема формирования штрих-кода:**

****

# 3.5 Проверка статуса созданного заказа в системе

Проверка статуса также осуществляется по принципу POST запроса, с указанием обязательных переменных.

**URL для отправки POST массива:**

<https://api.dostavka.guru/client/order_info_post.php>

Для передачи данных в формате JSON:

<https://api.dostavka.guru/client/order_info_post.php>?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *order\_number* | Номер заказа в Вашей информационной системе. Обязательно должен быть индивидуальным для каждого заказа, кроме услуги ЗАБОР | **Текстовый** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[order\_number] => no-0000025

)

**Пример ответа сервера:**

<?xml version="1.0" encoding="UTF-8"?>  
<root>  
<order>  
<order\_id>no-0000025</order\_id>

<track\_id>00A45FB8</track\_id>  
<uid>uid\_415e35a0-b292-11e4-8e8f-386077e5d2cc</uid>  
<location\_status>У курьера</location\_status>  
<order\_status>Перенос</order\_status>

<order\_status\_date\_time>12.11.2018 15:21:30</order\_status\_date\_time>

<geo\_date\_time>12.11.2018 15:22:30</geo\_date\_time>

<track\_number>RN1234534534234CN</track\_number>

<payment\_cashback>0</payment\_cashback>

<order\_status\_date\_time> 02.03.2019 16:17:05 </order\_status\_date\_time>

<geo\_date\_time> 02.03.2019 16:17:05</geo\_date\_time>

<ves\_kg>1.2</ves\_kg>

<payment\_cash>1265</payment\_cash>

<ocen\_sum>1265</ocen\_sum>

<nal\_plat>1265</nal\_plat>

<usluga>ДОСТАВКА</usluga>

<sposob\_dostavki>Курьер</sposob\_dostavki>

<payment\_cashless>1265</payment\_cashless>

<payment\_cash\_nal>0</payment\_cash\_nal>

<tarif\_hran>0</tarif\_hran>

<tarif\_vozv>0</tarif\_vozv>

<tarif\_kompl>0</tarif\_kompl>

<tarif\_ypakovka>0</tarif\_ypakovka>

<tarif\_ves>0</tarif\_ves>

<tarif\_prodl>0</tarif\_prodl>

<price\_agent\_reward>0</price\_agent\_reward>

<price\_sms>0</price\_sms>

<price\_lift>0</price\_lift>

<price\_delivery\_zone>0</price\_delivery\_zone>

<payment\_delivery>265</payment\_delivery>

<tracking\_history>

<track>

<date>22.03.2017</date>

<time>0:00:00</time>

<id>185877</id>

<zip>140961</zip>

<description>Прием; Партионный;</description>

</track>

<track>

<date>22.03.2017</date>

<time>7:29:54</time>

<id>185877</id>

<zip>140983</zip>

<description>Обработка; Сортировка;</description>

</track>

</tracking\_history>

<goods>

<product>

<name>Escentric 01</name>

<code>14930</code>

<characteristic/>

<amount>1</amount>

<unit>шт.</unit>

<price>0</price>

<desc>Возврат брак</desc>

<return>1</return>

</product>

<product>

<name>Escentric 02</name>

<code>14929</code>

<characteristic/>

<amount>1</amount>

<unit>шт.</unit>

<price>7000</price>

<desc/>

<return/>

</product>

</goods>

<barcodes>  
<code>9999000005515</code>  
<code>9999000005522</code>  
<code>9999000005539</code>  
</barcodes>  
</order>  
</root>

Где "order\_id" - номер заказа в Вашей информационной системе, "uid" - дополнительный идентификатор в системе dostavka.guru, "track\_id" – идентификатор трека для отслеживания посылки, "location\_status" - местоположение, "order\_status" - статус заказа, "barcodes" - массив штрих кодов, "track\_number " - идентификатор отправления, "tracking\_history" - массив отслеживания посылки, при отправке почтой, который состоит из элементов "track" с параметрами "date" - дата поступления, "time" - время поступления, "id" - идентификатор заказа, "zip" - индекс почтового отделения, "description" - описание операции, order\_status\_date\_time - дата и время последнего изменения статуса заказа, geo\_date\_time - дата и время последнего изменения местоположения, goods – блок потоварного учета, где return – количество возврата в позиции, desc – описание причины возврата, tarif\_hran - стоимость услуги хранения заказа на складе курьерской компании, tarif\_vozv - стоимость услуги возврата заказа интернет-магазину, tarif\_kompl - стоимость услуги по комплектации заказа, tarif\_ypakovka - стоимость услуги по упаковке заказа, tarif\_ves - тариф вес, tarif\_prodl - стоимость услуги по хранинию заказа на ПВЗ сверх срока договора, price\_agent\_reward - стоимость агентских услуг курьерской компании по заказу (прием денежных средств), price\_sms - стоимость услуги по СМС информированию получателя заказа о дате и времени доставки, price\_lift - стоимость услуги по подъему заказа весом свыше нормы, price\_delivery\_zone - тариф за доставку за пределы МКАД (КАД), payment\_cashless - сумма принятых денежных средств безналичным способом, payment\_cashback - стоимость возвратов, payment\_cash - сумма принятых денежных средств, payment\_cash\_nal - сумма принятых наличных денежных средств.

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
Error: 600 - Заказы не найдены!  
и прочие...

# 3.6 Получение ближайших 30 релевантных дат доставки

Получение валидных дат возможно с помощью POST запроса. В ответе будут представлены 30 валидных и ближайших дат, на которые можно поставить доставку в формате XML.

**URL для отправки POST массива:**

[*http://api.dostavka.guru/client/date\_dost.php*](http://api.dostavka.guru/client/date_dost.php)

Для передачи данных в формате JSON:

[*http://api.dostavka.guru/client/date\_dost.php*?json=yes](http://api.dostavka.guru/client/date_dost.php?json=yes)

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

)

**Пример ответа сервера:**

<?xml version="1.0" encoding="UTF-8"?>  
<root>  
<day>**2015.02.13**</day>  
<day>**2015.02.14**</day>  
<day>**2015.02.16**</day>  
<day>**2015.02.17**</day>  
<day>**2015.02.18**</day>  
<day>**2015.02.19**</day>  
<day>**2015.02.20**</day>  
<day>**2015.02.21**</day>  
<day>**2015.02.24**</day>  
<day>**2015.02.25**</day>  
<day>**2015.02.26**</day>  
<day>**2015.02.27**</day>  
<day>**2015.02.28**</day>  
<day>**2015.03.02**</day>  
<day>**2015.03.03**</day>  
<day>**2015.03.04**</day>  
<day>**2015.03.05**</day>  
<day>**2015.03.06**</day>  
<day>**2015.03.07**</day>  
<day>**2015.03.09**</day>  
<day>**2015.03.10**</day>  
<day>**2015.03.11**</day>  
<day>**2015.03.12**</day>  
<day>**2015.03.13**</day>  
<day>**2015.03.14**</day>  
<day>**2015.03.16**</day>  
<day>**2015.03.17**</day>  
<day>**2015.03.18**</day>  
<day>**2015.03.19**</day>  
<day>**2015.03.20**</day>  
</root>

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

*Error: 200 - Ключ API не верен!  
и прочие...*

# 3.7 Массовая проверка статусов заказа

**URL для отправки POST массива:**

http://api.dostavka.guru/client/orders\_info.php

Для передачи данных в формате JSON:

[http://api.dostavka.guru/client/orders\_info.php?json=yes](%20http://api.dostavka.guru/client/orders_info.php?json=yes)

Массив параметров:

POST:{

**['partner\_id']**=>'Ваш ID партнера'

**['key']**=>'Ваш ключ API'

**['orders']**=>список номеров заказов через ";" (через точку с запятой)

}

**Пример ответа сервера XML:**

<order>

<order\_number>**ch\_dost\_33\_lk**</order\_number>

<track\_id>00A45FB8</track\_id>

<status/>

<geo>**Ожидание**</geo>

<date\_dost>**2019-01-25**</ date\_dost >

<payment\_cashback>**0**</payment\_cashback>

<payment\_cash>**0**</payment\_cash>

<payment\_cash\_nal>**0**</payment\_cash\_nal>

<payment\_cashless>**0**</payment\_cashless>

<tarif\_hran>**0**</tarif\_hran>

<tarif\_vozv>**0**</tarif\_vozv>

<tarif\_kompl>**0**</tarif\_kompl>

<tarif\_ypakovka>**0**</tarif\_ypakovka>

<tarif\_ves>**0**</tarif\_ves>

<tarif\_prodl>**0**</tarif\_prodl>

<price\_agent\_reward>**0**</price\_agent\_reward>

<price\_sms>**0**</price\_sms>

<price\_lift>**0**</price\_lift>

<price\_delivery\_zone>**0**</price\_delivery\_zone>

</order>

<order>

<order\_number>**ch\_dost\_13\_lk**</order\_number>

<track\_id>00A45FB8</track\_id>

<status/>

<geo>**Ожидание**</geo>

<date\_dost>**2019-01-25**</ date\_dost >

<payment\_cashback>**0**</payment\_cashback>

<payment\_cash>**0**</payment\_cash>

<payment\_cashless>0</payment\_cashless>

<tarif\_hran>0</tarif\_hran>

<tarif\_vozv>0</tarif\_vozv>

<tarif\_kompl>0</tarif\_kompl>

<tarif\_ypakovka>0</tarif\_ypakovka>

<tarif\_ves>0</tarif\_ves>

<tarif\_prodl>0</tarif\_prodl>

<price\_agent\_reward>0</price\_agent\_reward>

<price\_sms>0</price\_sms>

<price\_lift>0</price\_lift>

<price\_delivery\_zone>0</price\_delivery\_zone>

</order>

***Описание полей ответа:***

**order\_number -** Номер заказа

**track\_id -** Идентификатор трека для отслеживания посылки

**status** - Статус заказа

**geo** - Местоположение заказа

**date\_dost** – Дата доставки

**payment\_cashback** - Сумма возврата

**payment\_cash** - Сумма прихода

**payment\_cash\_nal** - Сумма принятых наличных денежных средств

**payment\_cashless** - Сумма принятых безналичных денежных средств

**tarif\_hran** - Стоимость услуги хранения заказа на складе курьерской компании

**tarif\_vozv** - Стоимость услуги возврата заказа интернет-магазину

**tarif\_kompl** - Стоимость услуги по комплектации заказа

**tarif\_ypakovka** - Стоимость услуги по упаковке заказа

**tarif\_ves** - Тариф вес

**tarif\_prodl** - Стоимость услуги по хранению заказа на ПВЗ сверх срока договора

**price\_agent\_reward** - Стоимость агентских услуг курьерской компании по заказу (прием денежных средств)

**price\_sms** - Стоимость услуги по СМС информированию получателя заказа о дате и времени доставки

**price\_lift** - Стоимость услуги по подъему заказа весом свыше нормы

**price\_delivery\_zone** - Тариф за доставку за пределы МКАД (КАД)

# 3.8 Печать Акта передачи Заказов курьеру

**URL для отправки POST массива:**

http://api.dostavka.guru/client/barcode/print\_act\_pdf.php

Для передачи данных в формате JSON:

<http://api.dostavka.guru/client/barcode/print_act_pdf.php?json=yes>

**Массив параметров:**

POST:{

**['partner\_id']**=>'Ваш ID партнера'

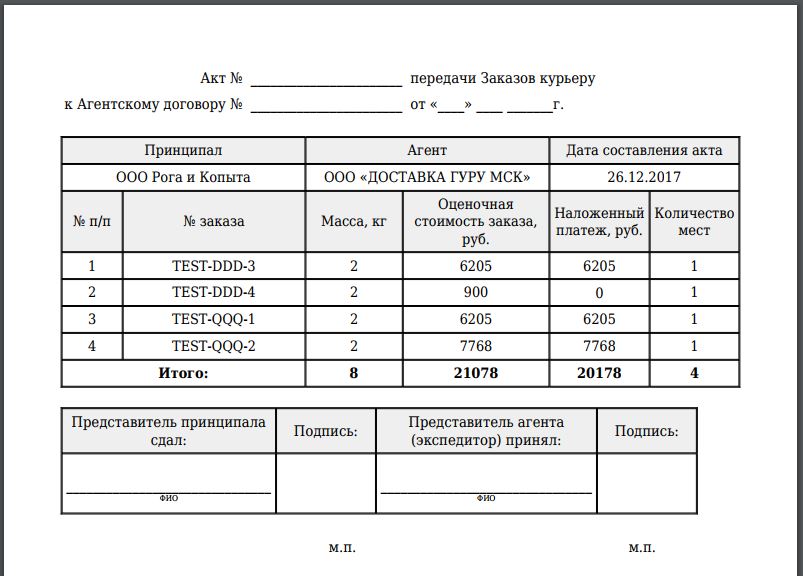
**['key']**=>'Ваш ключ API'

**['orders']**=>список номеров заказов через ";" (через точку с запятой)

**['print']**=>'y' (постоянная - всегда равна 'y')

}

**Пример ответа сервера PDF:**

****

# 3.9 Массовая печать штрих-кодов

**URL для отправки POST массива:**

http://api.dostavka.guru/client/barcode/print\_barcodes\_pdf.php

Для передачи данных в формате JSON:

<http://api.dostavka.guru/client/barcode/print_barcodes_pdf.php?json=yes>

**Массив параметров:**

POST:{

**['partner\_id']**=>'Ваш ID партнера'

**['key']**=>'Ваш ключ API'

**['orders']**=>список номеров заказов через ";" (через точку с запятой)

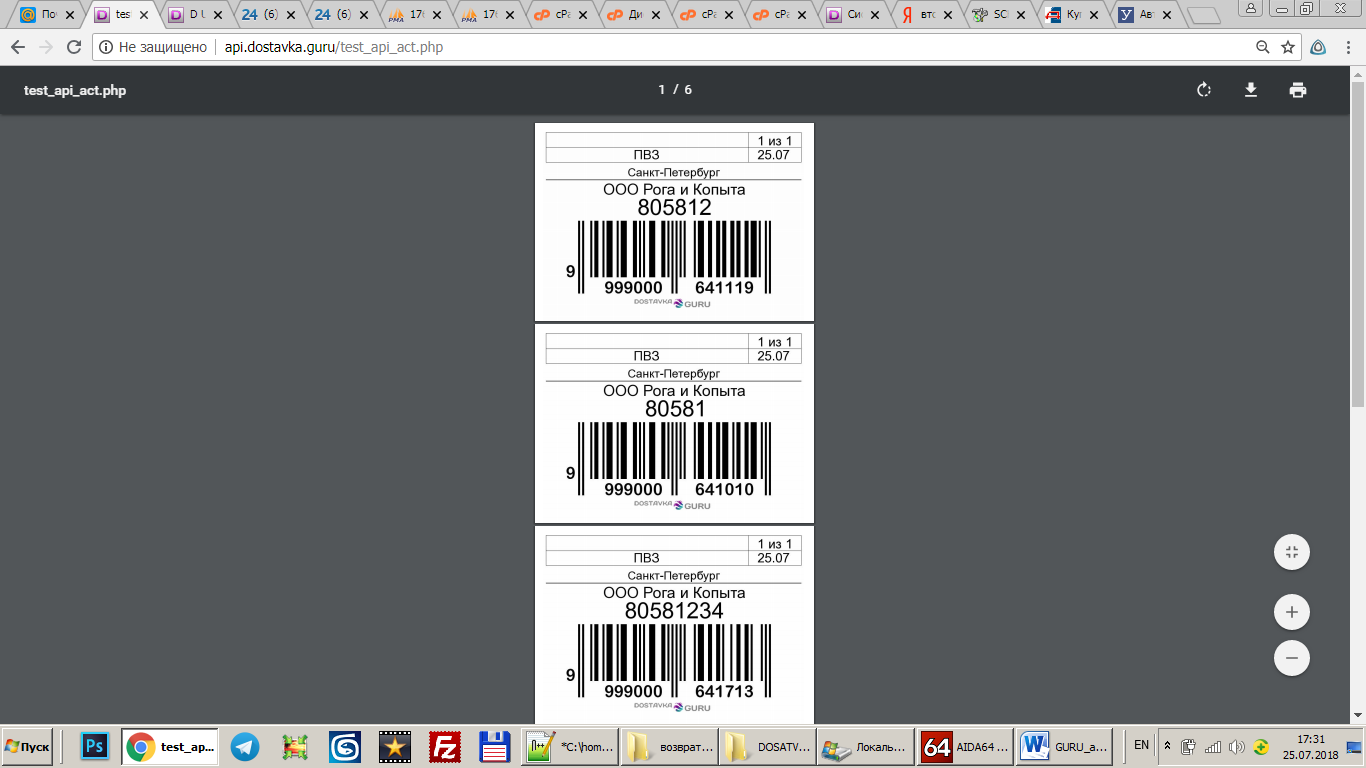
**['go\_print']**=>'**y**' (постоянная - всегда равна 'y')

**['print\_type']**=> тип печати "**list\_1**" - 1 штрих-код на одном листе, "**list\_18**" - 18 штрих-кодов на одном листе

**['start\_pos']**=> стартовая позиция печати (целое число)

}

**Пример ответа сервера PDF:**



# 3.10 Отправка разовой заявки на ЗАБОР

**URL для отправки POST массива:**

[https://api.dostavka.guru/client/in\_up\_2\_0.php](https://api.dostavka.guru/client/order_info_post.php)

Для передачи данных в формате JSON:

[https://api.dostavka.guru/client/in\_up\_2\_0.php](https://api.dostavka.guru/client/order_info_post.php)?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *usluga* | Наименование услуги | **“ЗАБОР”** |
| *warehouse\_id* | ID склада (см. п. 3.11) | **Строковый** |
| *date\_dost* | Дата заявки | **YYYY.MM.DD** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[usluga] => ЗАБОР

[warehouse\_id] => 095502e3-9b1b-11e8-84d1-000c294bb3ff

[date\_dost] => 2018.08.29

)

**Пример ответа сервера:**

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

<order>

<order\_number>ЗАБОР(new1535374432g9c)</order\_number>

<location\_status>Ожидание (Забор добавлен в систему)</location\_status>

<barcodes>

<code>9999000649511</code>

</barcodes>

</order>

</root>

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
Error: 600 - Заказы не найдены!  
и прочие...

# 3.11 Получение списка складов с идентификаторами

**URL для отправки POST массива:**

[https://api.dostavka.guru/client/warehouses.php](https://api.dostavka.guru/client/order_info_post.php)

Для передачи данных в формате JSON:

[https://api.dostavka.guru/client/warehouses.php](https://api.dostavka.guru/client/order_info_post.php)?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *script* | Наименование скрипта обработки данных | **warehouses** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[script] => warehouses

)

**Пример ответа сервера:**

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

<warehouse>

<warehouse\_id>123243a-3a57-11e7-950f-000c4754834</warehouse\_id>

<warehouse\_name>9999 Склад ДГ Новохохловская 21</warehouse\_name>

<city\_from>Москва</city\_from>

<organization>ДоставкаГУРУ</organization>

<cont\_tel>+79295552233234</cont\_tel>

<cont\_name>Василий</cont\_name>

<adres>Москва, Новохохловская, 23</adres>

<comment/>

<sms\_tel>+79999999999</sms\_tel>

<time\_s\_po>10:00-15:00</time\_s\_po>

<exp/>

</warehouse>  
</root>

Где "warehouse\_id" – идентификатор склада, "warehouse\_name" – имя склада, " city\_from" – город из, " organization" – наименование организации, "cont\_tel" – контактный телефон, "cont\_name" – ФИО для контактов, "adres" – адрес склада, " comment " – комментарий к складу, " sms\_tel " – телефон для SMS информирования, " time\_s\_po " - время доставки.

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
Error: 600 - Заказы не найдены!  
и прочие...

# 4. Дополнительно

Дополнительные сервисы для автоматизации обмена данными

# 4.1 Получение полного списка ПВЗ и списка пунктов с передачей заказов в курьерскую службу

Получение полного списка пунктов выдачи по принципу POST запроса, с указанием обязательных переменных. Список будет передан в XML формате. Будьте внимательны с точками выдачи заказов с полностью оплаченными заказами.

**URL для отправки POST массива:**

<https://api.dostavka.guru/client/pvz_list.php>

Для передачи данных в формате JSON:

<https://api.dostavka.guru/client/pvz_list.php>?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *script* | тип скрипта (**all\_list** - список ПВЗ, **wh\_list** – список ПВЗ с передачей заказов в курьерскую службу) | **Текстовый** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[script] => all\_list

)

**Пример ответа сервера:**

<?xml version="1.0" encoding="UTF-8"?>  
<root>  
<point>  
<city\_code>5</city\_code>  
<name>Абакан</name>  
<city\_name>Абакан</city\_name>  
<electronic\_aut>Да</electronic\_aut>  
<address>  
655017, г. Абакан, ул. Ленинского Комсомола, д. 35, офис №76Н  
</address>  
<tarif\_zone>6</tarif\_zone>  
<settlement>АБАКАН</settlement>  
<desc\_pvz>  
Район Черногорского парка, перейти регулируемый пешеходный переход и в правой жилой пятиэтажке находится наш офис. Второй офис слева.  
</desc\_pvz>  
<visa>Да</visa>  
<metro/>  
<work\_time>пн-пт:08.00-20.00 сб:10.00-16.00</work\_time>  
<telephone>8-800-700-54-30</telephone>  
<courier>Да</courier>  
<country>Россия</country>  
<delivery\_period>8</delivery\_period>  
<type/>  
<small\_address>ул. Ленинского Комсомола, д. 35, офис №76Н</small\_address>  
<area>ХАКАСИЯ РЕСПУБЛИКА</area>  
<only\_pay>Нет</only\_pay>  
<geo\_X>53.722047</geo\_X>  
<geo\_Y>91.429928</geo\_Y>  
<code>191</code>  
<delivery\_date>2015-06-25</delivery\_date>  
</point>  
<point>  
<city\_code>92</city\_code>  
<name>Альметьевск</name>  
<city\_name>Альметьевск</city\_name>  
<electronic\_aut>Да</electronic\_aut>  
<address> 423462, г. Альметьевск, ул. Маяковского, д. 47, пом. 2 </address>  
<tarif\_zone>4</tarif\_zone>  
<settlement>АЛЬМЕТЬЕВСК</settlement>  
<desc\_pvz>  
Остановка "Нефтяной институт", проезд автобусами №№5,8, троллейбусами №№ 1,2. От остановки "Нефтяной Институт" пройти 100 м и повернуть на ул. Маяковского в сторону "Таттелекома", спуститься вниз по ул. Маяковского, вдоль забора Нефтяного института, первый дом - отделение Боксберри, дверь с левого т  
</desc\_pvz>  
<visa>Да</visa>  
<metro/>  
<work\_time>пн-пт:10.00-19.00</work\_time>  
<telephone>8-800-700-54-30</telephone>  
<courier>Да</courier>  
<country>Россия</country>  
<delivery\_period>6</delivery\_period>  
<type/>  
<small\_address>ул. Маяковского, д. 47, пом. 2</small\_address>  
<area>ТАТАРСТАН РЕСПУБЛИКА</area>  
<only\_pay>Нет</only\_pay>  
<geo\_X>54.9053737</geo\_X>  
<geo\_Y>52.3108395</geo\_Y>  
<code>222</code>  
<delivery\_date>2015-06-23</delivery\_date>  
</point>  
</root>

**Описание полей:**

"code"=>"Код ПВЗ",  
"name"=>"Наименование ПВЗ",  
"address"=>"Полный адрес",  
"telephone"=>"Телефон или телефоны",  
"desc\_pvz"=>"Описание проезда",  
"delivery\_period"=>"Срок доставки",  
"city\_code"=>"Код города",  
"city\_name"=>"Наименование города",  
"tarif\_zone"=>"Тарифная зона",  
"settlement"=>"Населенный пункт",  
"area"=>"Регион",  
"country"=>"Страна",  
"geo\_X", "geo\_Y"=>"Координаты gps",  
"only\_pay"=>"Если значение "Да" - точка работает только с полностью оплаченными заказами",  
"small\_address"=>"Сокращенный адрес",  
"visa"=>"Если значение "Да" - Есть возможность оплаты платежными (банковскими) картами",  
"electronic\_aut"=>"Если значение "Да" - Подпись получателя будет хранится в системе в электронном виде",  
"metro"=>"Ближайшая станция Метро",  
"work\_time"=>"График работы",  
"delivery\_date"=>"Ближайшая дата доставки"

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
и прочие...

# 4.2 Получение информации о ПВЗ по его ID

Получение параметров ПВЗ по его ID по принципу POST запроса, с указанием обязательных переменных.

**URL для отправки POST массива:**

<https://api.dostavka.guru/client/pvz_list.php>

Для передачи данных в формате JSON:

<https://api.dostavka.guru/client/pvz_list.php>?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *script* | тип скрипта (**all\_list** - список ПВЗ) | **Текстовый** |
| *pointID* | код ПВЗ | **Текстовый** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[script] => all\_list

[pointID] => 19

)

**Пример ответа сервера:**

Ответ сервера такой же, как в пункте **4.1**

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

Error: 200 - Ключ API не верен!  
и прочие...

# 4.3 Получение списка городов с ПВЗ

Получение полного списка городов пунктов выдачи по принципу POST запроса, с указанием обязательных переменных.

**URL для отправки POST массива:**

[*https://api.dostavka.guru/client/pvz\_list.php*](https://api.dostavka.guru/client/pvz_list.php)

Для передачи данных в формате JSON:

<https://api.dostavka.guru/client/pvz_list.php>?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *script* | тип скрипта (**all\_city** - список городов ПВЗ) | **Текстовый** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[script] => all\_city

)

**Пример ответа сервера:**

<?xml version="1.0" encoding="UTF-8"?>  
<root>  
<city>  
<city\_code>5</city\_code>  
<city\_name>Абакан</city\_name>  
</city>  
<city>  
<city\_code>92</city\_code>  
<city\_name>Альметьевск</city\_name>  
</city>  
</root>

**Обработка ошибок:**

Если какие-то поля будут заполнены не правильно, то сервер выдаст сообщение об ошибке. К примеру:

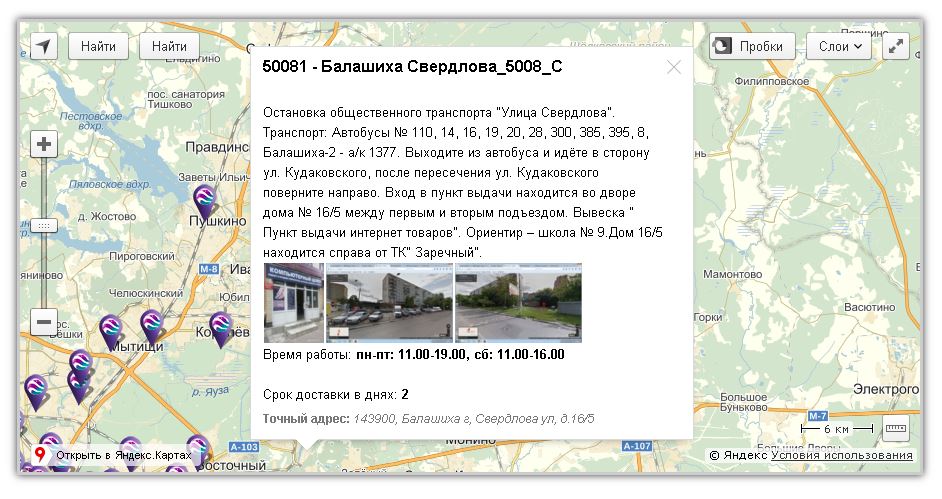
*Error: 200 - Ключ API не верен!  
и прочие...*

# 4.4 Вставка карты ПВЗ на сайт, как информационная страница

Данный пример позволит вставить карту в информационную страницу сайта для ознакомления клиентов с пунктами выдачи заказов.

Пример работы скрипта можно посмотреть здесь:

<https://api.dostavka.guru/client/map_with_but/map_simple.html>



**Как вставить код карты:**

Карта(Вставляется внутри тега "body" в нужном для Вас месте):

<div style="padding:10px; margin:10px; border:solid 1px #cccccc; width:300px;">

<div id="YMapsID" style="z-index:998; display:block; width: 900px; height: 450px; margin:10px; border:solid 3px #cccccc; right:50px;">

</div>

</div>

Скрипт инициализации карты(первая строчка не нужна, если jQuery уже подключен)(Вставляется внутри тега "head" на нужной для Вас странице):

<script src="https://api.dostavka.guru/jquery.js"></script>

<style>

#YMapsID{

box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

-moz-box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

-webkit-box-shadow: 0px 0px 12px rgba(0,0,0,0.8);

}

</style>

<script src="https://api.dostavka.guru/client/collection-search-provider.js"></script>

<script src="https://api-maps.yandex.ru/2.1/?load=package.standard,package.geoObjects&lang=ru-RU" type="text/javascript"></script>

<script src="https://api.dostavka.guru/client/map\_info.js"></script>

# 4.5 Проверка качества адреса

Получение варианта исправления адреса и код точности по принципу POST запроса, с указанием обязательных переменных.

**URL для отправки POST массива:**

[*https://api.dostavka.guru/client/check\_addr.php*](https://api.dostavka.guru/client/check_addr.php)

Для передачи данных в формате JSON:

[*https://api.dostavka.guru/client/check\_addr.php*](https://api.dostavka.guru/client/check_addr.php)?json=yes

**Переменные:**

|  |  |  |
| --- | --- | --- |
| Поле | Описание | Формат |
| *partner\_id* | Уникальный ID в системе dostavka.guru. | **Четырехзначный, числовой: 9999** |
| *key* | Уникальный API ключ в системе dostavka.guru. | **Хеш код** |
| *addr* | Адрес доставки | **Текстовый UTF-8** |

**Пример POST массива для отправки:**

$\_POST = array

(

[partner\_id] => 9999

[key] => cc03e747a6afbbcbf8be7668acfebee5

[addr] => егорьевск 1-й мкн 35 34

)

**Пример ответа сервера:**

Ответ сервера будет массив в формате json. Для извлечения массива используйте PHP функцию: *$array=json\_decode($data, true)*

Array

(

[structure] => Array

(

[0] => ADDRESS

)

[data] => Array

(

[0] => Array

(

[0] => Array

(

[source] => егорьевск 1-й мкн 35 34

[result] => Россия, Московская обл, г Егорьевск, мкр 1-й, д 35, кв 34

[postal\_code] => 140304

[country] => Россия

[region\_type] => обл

[region\_type\_full] => область

[region] => Московская

[area\_type] => р-н

[area\_type\_full] => район

[area] => Егорьевский

[city\_type] => г

[city\_type\_full] => город

[city] => Егорьевск

[settlement\_type] =>

[settlement\_type\_full] =>

[settlement] =>

[city\_district] =>

[street\_type] => мкр

[street\_type\_full] => микрорайон

[street] => 1-й

[house\_type] => д

[house\_type\_full] => дом

[house] => 35

[block\_type] =>

[block\_type\_full] =>

[block] =>

[flat\_type] => кв

[flat] => 34

[postal\_box] =>

[kladr\_id] => 5000700100000010030

[fias\_id] => 31158fcd-b058-4b80-a43b-e8dcb3c0057c

[capital\_marker] => 1

[okato] => 46212501000

[oktmo] => 46612101001

[tax\_office] => 5011

[tax\_office\_legal] =>

[flat\_area] => 32.1

[square\_meter\_price] =>

[flat\_price] =>

[timezone] => UTC+3

[geo\_lat] => 55.3738931

[geo\_lon] => 39.062796

[qc\_geo] => 0

[qc\_complete] => 0

[qc\_house] => 2

[qc] => 0

[unparsed\_parts] =>

)

)

)

)

Где параметр **qc** - отвечает за точность адреса. Нужно добиваться, чтобы адрес выдавал код точности "0". Все остальные система заявок DOSTAVKA.guru не пропустит.

**Описание остальных полей:**

| **Название** | **Длина** | **Описание** |
| --- | --- | --- |
| source | 250 | Исходный адрес одной строкой |
| result | 500 | Стандартизованный адрес одной строкой |
| postal\_code | 6 | Индекс |
| country | 120 | Страна |
| region\_type | 10 | Тип региона (сокращенный) |
| region\_type\_full | 50 | Тип региона |
| region | 120 | Регион |
| area\_type | 10 | Тип района в регионе (сокращенный) |
| area\_type\_full | 50 | Тип района в регионе |
| area | 120 | Район в регионе |
| city\_type | 10 | Тип города (сокращенный) |
| city\_type\_full | 50 | Тип города |
| city | 120 | Город |
| settlement\_type | 10 | Тип населенного пункта (сокращенный) |
| settlement\_type\_full | 50 | Тип населенного пункта |
| settlement | 120 | Населенный пункт |
| city\_district | 120 | Район города |
| street\_type | 10 | Тип улицы (сокращенный) |
| street\_type\_full | 50 | Тип улицы |
| street | 120 | Улица |
| house\_type | 10 | Тип дома (сокращенный) |
| house\_type\_full | 50 | Тип дома |
| house | 50 | Дом |
| block\_type | 10 | Тип корпуса/строения (сокращенный) |
| block\_type\_full | 50 | Тип корпуса/строения |
| block | 50 | Корпус/строение |
| flat\_type | 10 | Тип квартиры |
| flat | 50 | Квартира |
| flat\_area | 50 | Площадь квартиры |
| square\_meter\_price | 50 | Рыночная стоимость м? |
| flat\_price | 50 | Рыночная стоимость квартиры |
| postal\_box | 50 | Абонентский ящик |
| fias\_id | 36 | Код ФИАС:  HOUSE.HOUSEGUID, если дом найден в ФИАС по точному совпадению;  HOUSEINT.HOUSEGUID, если дом найден в ФИАС как часть интервала;  ADDROBJ.AOGUID в противном случае. |
| kladr\_id | 19 | Код КЛАДР |
| capital\_marker | 1 | Является ли город центром:  1 — центр района (Московская обл, Одинцовский р-н, г Одинцово)  2 — центр региона (Новосибирская обл, г Новосибирск);  3 — центр района и региона (Костромская обл, Костромской р-н, г Кострома);  0 — ни то, ни другое (Московская обл, г Балашиха). |
| okato | 11 | Код ОКАТО |
| oktmo | 11 | Код ОКТМО |
| tax\_office | 4 | Код ИФНС для физических лиц |
| tax\_office\_legal | 4 | Код ИФНС для организаций (не заполняется) |
| timezone | 10 | Часовой пояс |
| geo\_lat | 10 | Координаты: широта |
| geo\_lon | 10 | Координаты: долгота |
| qc\_geo | 5 | Код точности координат |
| qc\_complete | 5 | Код полноты |
| qc\_house | 5 | Код проверки дома |
| qc | 5 | Код качества |
| unparsed\_parts | 250 | Нераспознанная часть адреса. Для адреса *Москва, Митинская улица, 40к2с2 кв 12, просил доставку после обеда — утром похмелье* вернем *ПРОСИЛ, ДОСТАВКУ, ПОСЛЕ, ОБЕДА, УТРОМ, ПОХМЕЛЬЕ*. |