

Cordial saludo, a continuación, les indicaremos la documentación para la integración con Dropi.

La siguiente lista, es el listado de servicios que se pueden integrar.

- Login de usuario.
- Lista de Departamentos.
- Lista de Ciudades.
- Crear Orden.
- Obtener orden específica por ID de Dropi.
- Obtener orden específica por número de guía.
- Obtener el listado de órdenes.
- Obtener la guía en pdf.
- Obtener el Historial de Cartera.
- Obtener Transportadoras.
- Cotizador de Flete.

A continuación, comparto los datos para el Login del usuario:

- Url de Pruebas: <https://test-api.dropi.co/api/login>
- Url de Producción: <https://api.dropi.co/api/login>
- Método: POST
- Cabeceras: Content-Type: application/json

Se deben enviar los siguientes tres parámetros:

- email
- password
- white_brand_id

El valor del parámetro white_brand_id, siempre debe ser:

“df3e6b0bb66ceaadca4f84cbc371fd66e04d20fe51fc414da8d1b84d31d178de”, sin las comillas.

Si el login es exitoso, retornará una variable llamada “token”, el cual se usará para identificarse dentro de Dropi.

Se debe enviar como cabecera: **Authorization: Bearer TOKEN**

Adjunto Ejemplo del envío de los datos:

```
{
  "email": "usuario@gmail.com",
  "password": "clave",
  "white_brand_id": "df3e6b0bb66ceaadca4f84cbc371fd66e04d20fe51fc414da8d1b84d31d178de"
}
```

Adjunto ejemplo desde Postman:

```
{
  "email": "usuario@gmail.com",
  "password": "clave",
  "white_brand_id": "df3e6b0bb66ceaadca4f84cbc371fd66e04d20fe51fc414da8d1b84d31d178de"
}
```

Adjunto ejemplo de respuesta en caso exitoso:

```
1  {
2    "isSuccess": true,
3    "message": "Ha ingresado al sistema correctamente",
4    "status": 200,
5    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJpc3MiOiJodHRwOlwvXC8xMjcuMC4wLjE6ODAwMFwvYXBpXC9sb2dpbiIsImhhdCI6MTY1NTE2MwYyZWY5ZmQxNTgxMmZkZW5NzE1M2ExNGUwYjA0NzU0NmFhIn0.R_AK4y_NTI8MpvAoneUqvE2
6    "objects": {
7      "id": 12,
8      "name": "Fernando",
9      "surname": "romero",
10     "dni_type": null,
11     "dni": null,
12     "email": "usuario@gmail.com",
13     "birthday": null,
14     "phone": null,
15     "store_phone": null,
16     "notes": "null",
```

Adjunto ejemplo de respuesta en caso de que la contraseña o el usuario sea incorrecto:

Body Cookies Headers (12) Test Results

Pretty

Raw

Preview

Visualize

JSON

⌵



```
1  {
2    "isSuccess": false,
3    "status": 401,
4    "message": "La combinación de inicio de sesión / correo electrónico no es correcta, intente nuevamente."
5  }
```

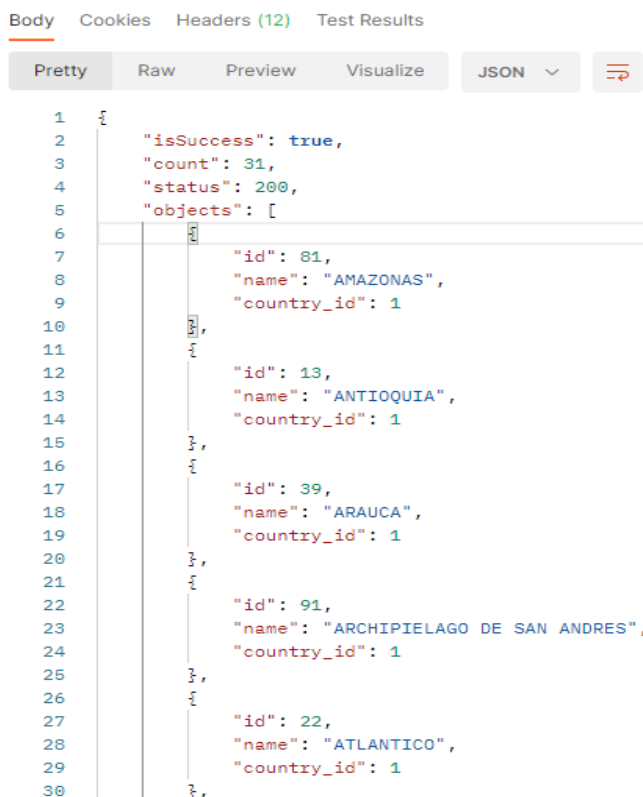
A continuación, comparto la respuesta del servidor en caso de que la sesión haya expirado, en ese caso, debe volver a hacer el Login:

```
{
  "isSuccess": false,
  "message": "Token is Expired",
  "status": 401
}
```

A continuación, comparto el api para obtener la lista de departamentos:

- Url de Prueba: <https://test-api.dropi.co/api/department>
- Url de Producción: <https://api.dropi.co/api/department>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

Adjunto un ejemplo de respuesta exitosa:



The screenshot shows a REST client interface with the 'Body' tab selected. The response is displayed in 'Pretty' format as JSON. The response indicates a successful status (200) and contains a list of five departments, each with an id, name, and country_id.

```
1 {
2   "isSuccess": true,
3   "count": 31,
4   "status": 200,
5   "objects": [
6     {
7       "id": 81,
8       "name": "AMAZONAS",
9       "country_id": 1
10    },
11    {
12      "id": 13,
13      "name": "ANTIOQUIA",
14      "country_id": 1
15    },
16    {
17      "id": 39,
18      "name": "ARAUCA",
19      "country_id": 1
20    },
21    {
22      "id": 91,
23      "name": "ARCHIPIELAGO DE SAN ANDRES",
24      "country_id": 1
25    },
26    {
27      "id": 22,
28      "name": "ATLANTICO",
29      "country_id": 1
30    }
31  ]
32 }
```

A continuación, comparto el api para obtener la lista de Ciudades:

- Url de Prueba: <https://test-api.dropi.co/api/trajectory/bycity>
- Url de Producción: <https://api.dropi.co/api/trajectory/bycity>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros:

- department_id:
 - tipo: int
 - Descripción: ID del departamento seleccionado
 - Requerido: SI
- rate_type:
 - tipo: string
 - Descripción: Para verificar si la ciudad tiene habilitado CON RECAUDO o SIN RECAUDO
 - Requerido: Si, se debe enviar vacío, si no se quiere especificar por un tipo de recaudo.
 - Ejemplo: CON RECAUDO o SIN RECAUDO. O enviar una cadena vacía con comillas.
 - Nota:

Adjunto un ejemplo de respuesta exitosa:

```
{
  "isSuccess": true,
  "count": 1227,
  "status": 200,
  "objects": [
    {
      "id": 523,
      "name": "ABEJORRAL",
      "department_id": 13,
      "department": {
        "id": 13,
        "country_id": 1,
        "name": "ANTIOQUIA",
        "created_at": "2020-02-17 21:46:12",
        "updated_at": "2020-02-17 21:46:12"
      },
      "rate_type": ["SIN RECAUDO"],
      "rate_type_json": [
        "SIN RECAUDO"
      ],
      "trajectory_type": "NACIONAL",
      "cod_dane": "5002000"
    },
    {
      "id": 524,
      "name": "ABREGO",
      "department_id": 17
    }
  ]
}
```

A continuación, comparto los datos para el proceso de Nueva Orden:

- Url de Prueba: <https://test-api.dropi.co/api/orders/myorders>
- Url de Producción: <https://api.dropi.co/api/orders/myorders>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros:

- state:
 - tipo: string
 - Descripción: Departamento al cual será enviada la orden
 - Requerido: SI
- city:
 - tipo: string
 - Descripción: Ciudad al cual será enviada la orden
 - Requerido: SI
- name:
 - tipo: string
 - Descripción: Nombres de cliente
 - Requerido: SI
- surname:
 - tipo: string
 - Descripción: Apellidos de cliente
 - Requerido: SI
- dir:
 - tipo: string
 - Descripción: Dirección
 - Requerido: SI
- client_email:
 - tipo: string
 - Descripción: Email
 - Requerido: NO
- notes:
 - tipo: string
 - Descripción: Nota que se quiera colocar en el rótulo y que se guarde en la orden
 - Requerido: NO
- payment_method_id:
 - tipo: int
 - Descripción: Método de pago, siempre con valor: 1
 - Requerido: SI
- phone:
 - tipo: string
 - Descripción: Teléfono celular del destinatario
 - Requerido: SI

- dni:
 - tipo: string
 - Descripción: Cédula o Dni, del cliente
 - Requerido: N
- dni_type:
 - tipo: string
 - Descripción: Tipo de identificación del cliente
 - Requerido: NO
- rate_type:
 - tipo: string
 - Descripción: Tipo de Pago, si va a ser “CON RECAUDO” o “SIN RECAUDO”
 - Requerido: SI
 - Ejemplo: CON RECAUDO
- type:
 - tipo: string
 - Descripción: Indica si la orden es una orden de prueba o no, (SIEMPRE DEBE IR FINAL_ORDER)
 - Requerido: SI
 - Ejemplo: FINAL_ORDER
- total_order:
 - tipo: int
 - Descripción: Total de la orden, incluyendo el costo del envío (precio de los productos x cantidad), sin comas ni puntos.
 - Requerido: SI
 - Ejemplo: 120000
- text_to_show_order_rotulo:
 - tipo: string
 - Descripción: Si este campo es distinto de null y distinto de vacío, entonces se mostrará lo que contenga este campo, en el rótulo que se crea en las guías, es decir, la información de los productos, cantidad, nombre, etc, se reemplazará por este texto.
 - Requerido: NO
 - Ejemplo: Texto para mostrar en el rótulo.
- external_warehouse_dir:
 - tipo: string
 - Descripción: Dirección del remitente, en caso de querer modificarlo. Esta dirección, aparecerá en la guía como dirección del remitente y desde donde debe salir el producto.
 - Requerido: NO
 - Ejemplo: cr 68 #19-65 barrio xxx
- external_warehouse_phone:
 - tipo: string

- Descripción: Teléfono del remitente, en el caso de querer modificar el remitente.
- Requerido: NO
- Ejemplo: 3115668987
- external_warehouse_city:
 - tipo: int
 - Descripción: ID de la ciudad en el caso de querer modificar el remitente. El ID debe ser del web service de Dropi de ciudades
 - Requerido: NO
 - Ejemplo: 12
- external_warehouse_contact_name:
 - tipo: string
 - Descripción: Indica el nombre de un contacto de la bodega, en la cual se despachará el producto, en caso de modificar los datos del remitente.
 - Requerido: NO
 - Ejemplo: Juan
- external_warehouse_contact_surname:
 - tipo: string
 - Descripción: Indica el nombre de un contacto de la bodega, en la cual se despachará el producto, en caso de modificar los datos del remitente.
 - Requerido: SI
 - Ejemplo: Perez
- **distributionCompany:**
 - tipo: object
 - Descripción: Indica si se quiere crear el pedido, con una transportadora específica. Si este parámetro no se envía, entonces tomará el orden de las parametrizaciones de transportadoras del usuario.
 - Requerido: NO
 - Índices:
 - id:
 - tipo: int
 - Descripción: ID de la transportadora. Para obtener el id de la transportadora, se puede consultar el web service de: **Obtener Transportadoras.**
 - Requerido: SI
- **products:**
 - tipo: object
 - Descripción: Objeto del producto o de los productos de la orden.
 - Requerido: SI
 - Índices:
 - id:
 - tipo: int
 - Descripción: ID del producto de Dropi.
 - Requerido: SI

- price:
 - tipo: int
 - Descripción: Precio de una unidad del producto, por ejemplo, si en la orden, son 2 unidades, y el total son 100000, entonces aquí debe ir 50000, ya que es solo el precio de una unidad.
 - Requerido: SI
- quantity:
 - tipo: int
 - Descripción: Cantidad a vender
 - Requerido: SI
- variation_id:
 - tipo: int
 - Descripción: Id Variación del producto
 - Requerido: NO si el producto no es variable.

El objeto “products”, es un objeto que cada sub índice, tendrá los datos de cada producto.

Adjunto un ejemplo del proceso de crear orden:

```
{
  "calculate_costs_and_shipping": true,
  "state": "CUNDINAMARCA",
  "city": "BOGOTA",
  "client_email": "fernandoga22@gmail.com",
  "name": "fernando",
  "surname": "perez",
  "dir": "cr 3 #2-09",
  "notes": "",
  "payment_method_id": 1,
  "phone": 313523645,
  "rate_type": "CON RECAUDO",
  "type": "FINAL_ORDER",
  "total_order": 260000,
  "products": [
    {
      "id": 1002,
      "price": 80000,
      "variation_id": null,
      "quantity": 2
    },
    {
      "id": 30100,
      "price": 50000,
      "variation_id": null,
      "quantity": 2
    }
  ]
}
```



```

1  [
2      "state": "CUNDINAMARCA",
3      "city": "BOGOTA",
4      "client_email": "fernandoga22@gmail.com",
5      "name": "fernando",
6      "surname": "perez",
7      "dir": "cr 3 #2-09",
8      "notes": "",
9      "payment_method_id": 1,
10     "phone": 313523645,
11     "rate_type": "CON RECAUDO",
12     "type": "FINAL_ORDER",
13     "total_order": 260000,
14     "products": [
15         {
16             "id": 1002,
17             "price": 80000,
18             "variation_id": null,
19             "quantity": 2
20         },
21         {
22             "id": 30100,
23             "price": 50000,
24             "variation_id": null,
25             "quantity": 2
26         }
27     ]
28 ]
29

```

Adjunto un ejemplo de respuesta exitosa:


```

{
  "isSuccess": true,
  "message": "El registro ha sido creado con exito!",
  "status": 200,
  "objects": {
    "state": "CUNDINAMARCA",
    "city": "BOGOTA",
    "client_email": "fernandoga22@gmail.com",
    "name": "fernando",
    "surname": "perez",
    "dir": "cr 38b #5-4",
    "notes": "por la derecha",

```

```
    "payment_method_id": 1,
    "phone": "313523645",
    "rate_type": "CON RECAUDO",
    "type": "FINAL_ORDER",
    "total_order": 260000,
    "overload_was_applied": true,
    "shipping_amount": 31460,
    "dropshipper_amount_to_win": 224540,
    "status": "PENDIENTE",
    "country": "COLOMBIA",
    "supplier_id": 3445,
    "user_id": 3445,
    "commision_supplier_make_own_order": "0.00",
    "updated_at": "2021-12-07 10:59:52",
    "created_at": "2021-12-07 10:59:52",
    "id": 523794,
    "discounted_amount": 31460
  },
  "wallets": [
    {
      "id": 3431,
      "user_id": 3445,
      "currency_id": 1,
      "amount": "405620.00",
      "created_at": "2021-12-06 10:09:07",
      "updated_at": "2021-12-07 10:59:52",
      "deleted_at": null
    }
  ]
}
```

Adjunto un ejemplo de respuesta en caso de que falte ingresar un campo requerido:



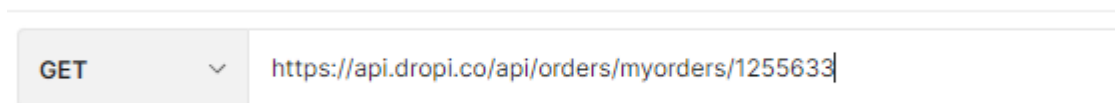
The screenshot shows the 'Body' tab of a web browser's developer tools. The response is a JSON object indicating an error. The JSON is formatted as follows:

```
1 {
2   "isSuccess": false,
3   "message": "Exception: Undefined property: stdClass::$surname",
4   "status": 400,
5   "error": "Undefined property: stdClass::$surname",
6   "file": "C:\\Users\\user\\PhpstormProjects\\backend\\app\\Http\\Controllers\\OrderController.php",
7   "line": 1607
8 }
```

A continuación, comparto el api para obtener la información de una orden específica por Orden ID de Dropi:

- Url de Prueba: <https://test-api.dropi.co/api/orders/myorders/IDORDEN>
- Url de Producción: <https://api.dropi.co/api/orders/myorders/IDORDEN>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

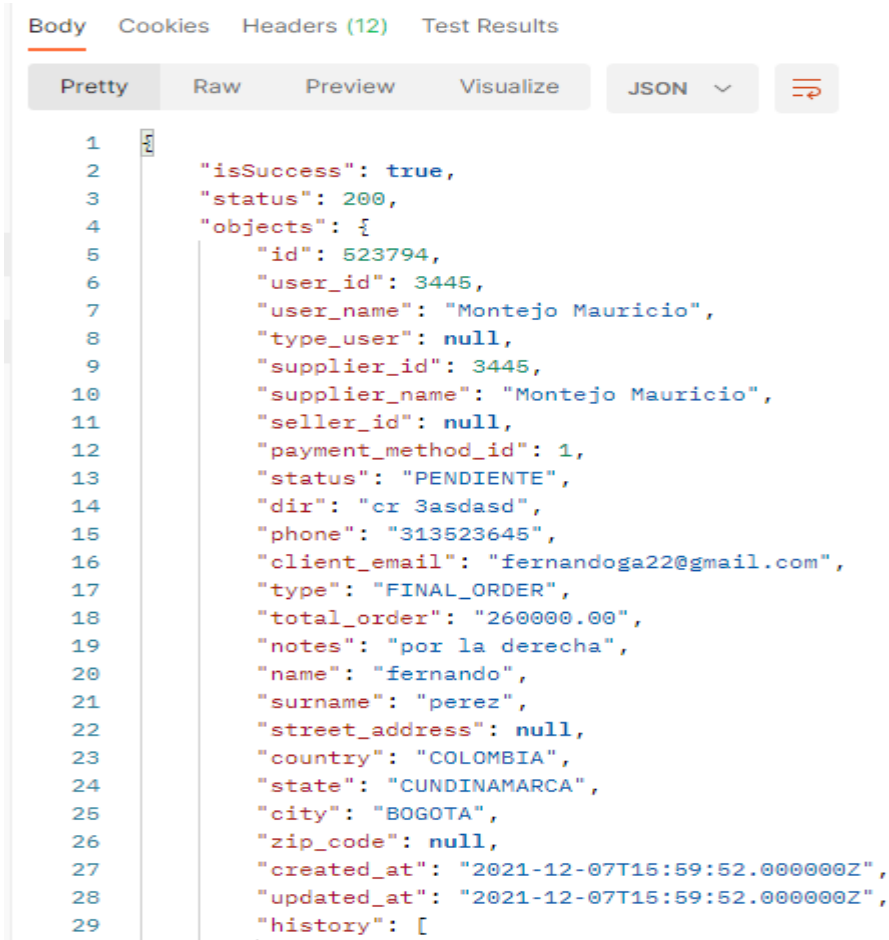
Adjunto ejemplo de la ejecución del api:



The screenshot shows a REST client interface with a 'GET' method selected and the following URL entered:

```
GET https://api.dropi.co/api/orders/myorders/1255633|
```

Adjunto un ejemplo de respuesta exitosa:



The screenshot shows a web browser's developer tools with the 'Body' tab selected. The response is a JSON object. The interface includes tabs for 'Body', 'Cookies', 'Headers (12)', and 'Test Results'. Below these are options for 'Pretty', 'Raw', 'Preview', and 'Visualize', along with a 'JSON' dropdown and a refresh icon. The JSON data is displayed in a syntax-highlighted, pretty-printed format with line numbers on the left.

```
1 {
2   "isSuccess": true,
3   "status": 200,
4   "objects": {
5     "id": 523794,
6     "user_id": 3445,
7     "user_name": "Montejo Mauricio",
8     "type_user": null,
9     "supplier_id": 3445,
10    "supplier_name": "Montejo Mauricio",
11    "seller_id": null,
12    "payment_method_id": 1,
13    "status": "PENDIENTE",
14    "dir": "cr 3asdasd",
15    "phone": "313523645",
16    "client_email": "fernandoga22@gmail.com",
17    "type": "FINAL_ORDER",
18    "total_order": "260000.00",
19    "notes": "por la derecha",
20    "name": "fernando",
21    "surname": "perez",
22    "street_address": null,
23    "country": "COLOMBIA",
24    "state": "CUNDINAMARCA",
25    "city": "BOGOTA",
26    "zip_code": null,
27    "created_at": "2021-12-07T15:59:52.000000Z",
28    "updated_at": "2021-12-07T15:59:52.000000Z",
29    "history": [
```

Ejemplo en formato json:

```
{
  "isSuccess": true,
  "status": 200,
  "objects": {
    "id": 523794,
    "user_id": 3445,
    "user_name": "Montejo Mauricio",
    "type_user": null,
    "supplier_id": 3445,
    "supplier_name": "Montejo Mauricio",
    "seller_id": null,
    "payment_method_id": 1,
    "status": "PENDIENTE",
    "dir": "cr 3asdasd",
    "phone": "313523645",
```

```

"client_email": "fernandoga22@gmail.com",
"type": "FINAL_ORDER",
"total_order": "260000.00",
"notes": "por la derecha",
"name": "fernando",
"surname": "perez",
"street_address": null,
"country": "COLOMBIA",
"state": "CUNDINAMARCA",
"city": "BOGOTA",
"zip_code": null,
"created_at": "2021-12-07T15:59:52.000000Z",
"updated_at": "2021-12-07T15:59:52.000000Z",
"history": [
  {
    "id": 2681960,
    "user_id": 3445,
    "order_id": 523794,
    "status": "PENDIENTE",
    "novedad_servientrega": null,
    "created_at": "2021-12-07T15:59:52.000000Z",
    "user": {
      "id": 3445
    }
  }
],
"shipping_guide": null,
"shipping_company": null,
"discounted_amount": "31460.00",
"shipping_amount": "31460.00",
"rate_type": "CON RECAUDO",
"supplier_amount": null,
"comission_percent": null,
"amount_earned_dropshipper": null,
"amount_earned_supplier": null,
"amount_earned_influencer": null,
"dropshipper_amount_to_win": "224540.00",
"sticker": null,
"guide_was_downloaded": false,
"overload_was_applied": true,
"novedad_servientrega": null,
"total_dropi_commission_product_supplier": null,
"total_dropi_commission_product_drops": null,
"total_commission_product_supplier": null,
"paymentmethod": {
  "id": 1,
  "name": "CARTERA",
  "available": true
},
"orderdetails": [
  {
    "id": 537379,
    "order_id": 523794,
    "price": "80000.00",
    "product": {
      "id": 14166,
      "id_lista": 0,
      "name": "Plato Grande XL",
      "name_in_order": "Plato Grande XL - $80000.00 - Stock 94.00",

```

```

        "description": null,
        "type": "SIMPLE",
        "stock": "94.00",
        "sale_price": "1000.00",
        "suggested_price": "80000.00",
        "user_id": 3445,
        "privated_product": true,
        "active": true,
        "sku": "7M18QCEYTU",
        "weight": "1.00",
        "length": "1.00",
        "width": "1.00",
        "height": "1.00",
        "deleted_at": null,
        "created_at": "2021-12-06T21:54:24.000000Z",
        "updated_at": null,
        "is_warehouse_dropi": false,
        "dropi_percent_product_supplier": null,
        "dropi_percent_product_drops": null,
        "user": {
            "id": 3445,
            "status": "Activo",
            "created_by": 22
        },
        "attributes": [],
        "variations": [],
        "gallery": [],
        "categories": [],
        "warehouses": [
            {
                "id": 489,
                "name": null,
                "user_id": 3445,
                "user": {
                    "id": 3445,
                    "status": "Activo",
                    "created_by": 22
                },
                "city_id": null,
                "city": null,
                "address": null,
                "phone": null,
                "warehouse_dropi": false,
                "created_at": null
            }
        ],
        "private_product_inventories": [],
        "user_profit_percentage_product": [],
        "persistanceState": "Unchanged",
        "shops": [],
        "external_code": "51"
    },
    "variation_id": null,
    "quantity": "2.00",
    "supplier_price": "1000.00",
    "shipping_amount": "19360.00",
    "price_increase_amount": "1000.00",
    "dropi_percent_product_supplier": "0.00",
    "total_dropi_commission_product_supplier": null,

```

```

"commission_product_supplier": "0.00",
"total_commission_product_supplier": null,
"drops_percent_product_drops": "5.00",
"total_drops_commission_product_drops": null,
"had_custom_percent_drop": false,
"had_custom_percent_supplier": false,
"amount_earned_dropshipper": null,
"amount_earned_supplier": null,
"amount_earned_influencer": null,
"user_profit_percentage_product": "0.00",
"separated_stock": "0.00",
"unseparated_stock": "2.00",
"private_stock": "0.00",
"commission_supplier_make_own_order": "0.00",
"variation": null
},
{
  "id": 537380,
  "order_id": 523794,
  "price": "50000.00",
  "product": {
    "id": 14167,
    "id_lista": 0,
    "name": "Plato Verde",
    "name_in_order": "Plato Verde - $50000.00 - Stock 94.00",
    "description": null,
    "type": "SIMPLE",
    "stock": "94.00",
    "sale_price": "1000.00",
    "suggested_price": "50000.00",
    "user_id": 3445,
    "privated_product": true,
    "active": true,
    "sku": "OKFRBAH4M9",
    "weight": "1.00",
    "length": "1.00",
    "width": "1.00",
    "height": "1.00",
    "deleted_at": null,
    "created_at": "2021-12-06T21:55:07.000000Z",
    "updated_at": null,
    "is_warehouse_drops": false,
    "drops_percent_product_supplier": null,
    "drops_percent_product_drops": null,
    "user": {
      "id": 3445,
      "status": "Activo",
      "created_by": 22
    }
  },
  "attributes": [],
  "variations": [],
  "gallery": [],
  "categories": [],
  "warehouses": [
    {
      "id": 489,
      "name": null,
      "user_id": 3445,
      "user": {

```

```

        "id": 3445,
        "status": "Activo",
        "created_by": 22
    },
    "city_id": null,
    "city": null,
    "address": null,
    "phone": null,
    "warehouse_dropi": false,
    "created_at": null
}
],
"private_product_inventories": [],
"user_profit_percentage_product": [],
"persistanceState": "Unchanged",
"shops": [],
"external_code": "52"
},
"variation_id": null,
"quantity": "2.00",
"supplier_price": "1000.00",
"shipping_amount": "12100.00",
"price_increase_amount": "1000.00",
"drops_percent_product_supplier": "0.00",
"total_dropi_commission_product_supplier": null,
"commission_product_supplier": "0.00",
"total_commission_product_supplier": null,
"drops_percent_product_drops": "5.00",
"total_dropi_commission_product_drops": null,
"had_custom_percent_drop": false,
"had_custom_percent_supplier": false,
"amount_earned_dropshipper": null,
"amount_earned_supplier": null,
"amount_earned_influencer": null,
"user_profit_percentage_product": "0.00",
"separated_stock": "0.00",
"unseparated_stock": "2.00",
"private_stock": "0.00",
"commission_supplier_make_own_order": "0.00",
"variation": null
}
],

```


A continuación, comparto el api para obtener la información de una orden específica por Número de guía específica:

- Url de Prueba: <https://test-api.dropi.co/api/orders/myorderbyguide/GUIA>
- Url de Producción: <https://api.dropi.co/api/orders/myorderbyguide/GUIA>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

Adjunto ejemplo de la ejecución del api:

GET	▼	https://api.dropi.co/api/orders/myorderbyguide/7290346973
-----	---	---

La respuesta es la misma que la documentación del api anterior.

```
1  {
2    "isSuccess": true,
3    "status": 200,
4    "objects": {
5      "id": 1537386,
6      "user_id": 3252,
7      "user_name": "Fernando Perez",
8      "type_user": null,
9      "supplier_id": 3252,
10     "supplier_name": "Fernando Perez",
11     "supplier": {
12       "id": 3252,
13       "name": "Fernando",
14       "surname": "Perez",
```

A continuación, comparto el api para obtener el listado de tus órdenes:

- Url de Prueba: <https://test-api.dropi.co/api/orders/myorders>
- Url de Producción: <https://api.dropi.co/api/orders/myorders>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

Parámetros:

- result_number:
 - tipo: int
 - Descripción: Cantidad de resultados (órdenes) que quieres que te devuelva el api
 - Requerido: NO (pero es recomendable hacerlo).
 - Ejemplo: 10
- start:
 - tipo: int
 - Descripción: Valor inicial desde el cual se quiere obtener los resultados.
 - Requerido: NO
 - Ejemplo: 1
- textToSearch:
 - tipo: string
 - Descripción: Texto, el cual buscará si coincide con: estatus, nombre de cliente, apellidos del cliente, dirección, departamento, ciudad, nombre de producto.
 - Requerido: NO
- from:
 - tipo: string
 - Descripción: Filtro para filtrar por fecha, este campo es “Desde” la fecha en la que se quiere filtrar. El formato es: yyyy-mm-dd
 - Requerido: NO
 - Ejemplo: 2021-12-07
- untill:
 - tipo: string
 - Descripción: Filtro para filtrar por fecha, este campo es “Hasta” la fecha en la que se quiere filtrar. El formato es: yyyy-mm-dd
 - Requerido: NO
 - Ejemplo: 2021-12-08
- radio_downloaded:
 - tipo: string
 - Descripción: Indica si quiere obtener las guías “IMPRESAS”, “NO IMPRESAS”, y las “IMPRESAS Y NO IMPRESAS”.
 - Requerido: NO
 - Posibles Valores: IMPRESAS Y NO IMPRESAS/ NO IMPRESAS/IMPRESAS

- Ejemplo: IMPRESAS Y NO IMPRESAS
- orderBy:
 - tipo: string
 - Descripción: Ordenar Por. Por defecto id
 - Requerido: NO
 - Ejemplo: id
- orderDirection:
 - tipo: string
 - Descripción: Ordenar de manera ascendente o descendente.
 - Requerido: NO
 - Ejemplo: desc
- filter_by:
 - tipo: string
 - Descripción: Filtra por una de las tres siguientes opciones específicas, y trae donde coincida con ese valor. Las opciones son: GUIA, ORDEN ID, CELULAR. Va acompañado con el siguiente parámetro value_filter_by.
 - Requerido: NO
 - Ejemplo: GUIA
- value_filter_by:
 - tipo: string
 - Descripción: Dependiendo de la opción que hayas enviado en el campo filter_by, buscará lo que coincida con el valor en ese parámetro. Puedes enviar varios resultados a la vez, separándolo por coma “,” o salto de línea, ejemplo: 014117806991, 014117806987.
 - Requerido: NO
 - Ejemplo: GUIA
- status:
 - tipo: string
 - Descripción: Para poder filtrar por estatus.
 - Requerido: NO
 - Ejemplo: PENDIENTE, GUIA_GENERADA
- filter_date_by:
 - tipo: string
 - Descripción: Indica, según el rango de fecha ingresado, que valor de fecha tomará. Hay tres valores posibles: “FECHA DE CREADO”, “FECHA DE PRIMERA IMPRESION”, “FECHA DE CAMBIO DE ESTATUS”.
 - FECHA DE CREADO: indica que filtrará por las órdenes que fueron creadas en ese rango de fechas.
 - FECHA DE PRIMERA IMPRESIÓN: indica que filtrará por las órdenes que fueron impresas la primera vez, en el rango de fecha.
 - FECHA DE CAMBIO DE ESTATUS: indica que filtrará por las órdenes que cambiaron a cierto estatus seleccionado, en ese rango de fechas.
 - Requerido: SI
 - Ejemplo: FECHA DE CREADO

Adjunto ejemplo en Postman:

```
{
  "from": "2022-05-23",
  "until": "2022-06-21",
  "status": "PENDIENTE"
}
```

Adjunto ejemplo de caso exitoso:

Body Cookies Headers (12) Test Results

Pretty Raw Preview Visualize JSON ↕

```
1  {
2    "isSuccess": true,
3    "status": 200,
4    "count": 173,
5    "objects": [
6      {
7        "id": 590375,
8        "user_id": 3747,
9        "supplier_id": 3747,
10       "payment_method_id": 1,
11       "status": "CANCELADO",
12       "dir": "calle 23 #7-84 12 de octubre",
13       "phone": "3128952713",
14       "created_at": "2021-12-11 08:05:02",
15       "updated_at": "2021-12-11 08:11:19",
16       "type": "FINAL_ORDER",
17       "total_order": "34700.00",
18       "notes": null,
19       "name": "Miller VILLASMI",
20       "surname": "Miller VILLASMI",
21       "street_address": null,
22       "country": "COLOMBIA",
23       "state": "LA GUAJIRA",
24       "city": "FONSECA",
25       "zip_code": null,
26       "comission_percent": null,
27       "amount_earned_dropshipper": null,
28       "amount_earned_supplier": null,
29       "shipping_amount": "10813.00",
30       "rate_type": "CON RECAUDO",
31       "shipping_guide": null,
32       "shipping_company": null,
33       "discounted_amount": "10813.00",
34       "sticker": null,
35       "guide_was_downloaded": false,
36       "overload_was_applied": false,
```

A continuación, comparto el api para generar la guía, a una orden específica:

- Url de Prueba: <https://test-api.dropi.co/api/orders/myorders/IDORDEN>
- Url de Producción: <https://api.dropi.co/api/orders/myorders/IDORDEN>
- Método: PUT
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

Parámetros:

- status:
 - tipo: string
 - Descripción: Estatus de Guia Generada para generar la guía.
 - Ejemplo: GUIA_GENERADA
 - Requerido: SI

Adjunto ejemplo de Postman:

```
{  
  "status": "GUIA_GENERADA"  
}
```

A continuación, comparto el api para generar guías de forma masiva:

- Url de Prueba: <https://test-api.dropi.co/api/orders/myorder/masive>
- Url de Producción: <https://api.dropi.co/api/orders/myorder/masive>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Debe ser un arreglo en el que cada índice del arreglo, pertenezca a una orden de Dropi especifica.

Parámetros:

- id:
 - tipo: integer
 - Descripción: Id de la orden en Dropi.
 - Ejemplo: 2008350
 - Requerido: SI
- status:
 - tipo: string
 - Descripción: Estatus de Guía Generada para generar la guía.
 - Ejemplo: GUIA_GENERADA
 - Requerido: SI

Adjunto ejemplo en Postman:

```
[  
  {  
    "id": 2008350,  
    "status": "GUIA_GENERADA"  
  },  
  {  
    "id": 2008280,  
    "status": "GUIA_GENERADA"  
  }  
]
```

Adjunto respuesta en caso de ser exitoso:

```
{  
  "isSuccess": true,  
  "message": "Actualizacion masiva lista",  
  "objects": [],  
  "status": 200  
}
```

A continuación, comparto la url para obtener el pdf de las guías:

- Url de Prueba: https://test-api.dropi.co/guias/nombre_transportadora/VARIABLE-STICKER
- Url de Producción: https://api.dropi.co/guias/nombre_transportadora/VARIABLE-STICKER
- Método: GET

La variable sticker, viene en el listado de órdenes, el parámetro llamado “**sticker**”, el cual contiene el nombre del archivo.

Ojo, para las guías con **Servientrega** y **Envia**, la url es <https://api.dropi.co/guias/servientrega/VARIABLE-STICKER>

Para las demás transportadoras, debe ir el nombre de la transportadora, ejemplo:

<https://api.dropi.co/guias/interrapidisimo/VARIABLE-STICKER>

<https://api.dropi.co/guias/coordinadora/VARIABLE-STICKER>

<https://api.dropi.co/guias/domina/VARIABLE-STICKER>

A continuación, comparto la url para obtener el historial de cartera del usuario:

- Url de Prueba: <https://test-api.dropi.co/api/historywallet>
- Url de Producción: <https://api.dropi.co/api/historywallet>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros:

- result_number:
 - tipo: int
 - Descripción: Cantidad de resultados (órdenes) que quieres que te devuelva el api
 - Requerido: NO (pero es recomendable hacerlo).
 - Ejemplo: 10
- start:
 - tipo: int
 - Descripción: Valor inicial desde el cual se quiere obtener los resultados.
 - Requerido: NO
 - Ejemplo: 1

- textToSearch:
 - tipo: string
 - Descripción: Texto, el cual buscará si coincide con: estatus, nombre de cliente, apellidos del cliente, dirección, departamento, ciudad, nombre de producto.
 - Requerido: NO
- from:
 - tipo: string
 - Descripción: Filtro para filtrar por fecha, este campo es “Desde” la fecha en la que se quiere filtrar. El formato es: yyyy-mm-dd
 - Requerido: NO
 - Ejemplo: 2021-12-07
- untill:
 - tipo: string
 - Descripción: Filtro para filtrar por fecha, este campo es “Hasta” la fecha en la que se quiere filtrar. El formato es: yyyy-mm-dd
 - Requerido: NO
 - Ejemplo: 2021-12-08
- orderBy:
 - tipo: string
 - Descripción: Ordenar Por. Por defecto id
 - Requerido: NO
 - Ejemplo: id
- orderDirection:
 - tipo: string
 - Descripción: Ordenar de manera ascendente o descendente.
 - Requerido: NO
 - Ejemplo: desc
- type:
 - tipo: string
 - Descripción: Indica si es ENTRADA o SALIDA, el valor puede ir en null, si no se quiere filtrar por un tipo específico.
 - Requerido: NO

A continuación, comparto el api para obtener la lista de Transportadoras:

- Url de Prueba: https://test-api.dropi.co/api/distribution_companies
- Url de Producción: https://api.dropi.co/api/distribution_companies
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN
- Nota: la url es: "api/distribution_companies" con el "_" en distribution_companies

Adjunto un ejemplo de respuesta exitosa:

```
{
  "isSuccess": true,
  "status": 200,
  "objects": [
    {
      "id": 1,
      "name": "SERVIENTREGA",
      "created_at": "2022-05-14 15:03:05",
      "updated_at": null,
      "deleted_at": null,
      "incidence_solution_method": "MEC2",
      "is_visible": true,
      "last_order_id_cron_update_status": null
    },
    {
      "id": 2,
      "name": "ENVIA",
      "created_at": "2022-05-14 15:03:05",
      "updated_at": "2022-09-26 11:30:21",
      "deleted_at": null,
      "incidence_solution_method": "URL",
      "is_visible": false,
      "last_order_id_cron_update_status": null
    }
  ]
}
```

A continuación, comparto la url para obtener el cotizador de flete:

- Url de Prueba: <https://test-api.dropi.co/api/orders/cotizaEnvioTransportadoraV2>
- Url de Producción: <https://api.dropi.co/api/orders/cotizaEnvioTransportadoraV2>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Para obtener el código dane de cada ciudad, puede usar el web service de **Lista de departamentos y Lista de Ciudades**. Para obtener las transportadoras disponibles, debe usar el web service de **Lista de Transportadoras**.

Este web service, retorna el costo de los fletes, con todas las transportadoras.

Parámetros:

- EnvioConCobro:
 - tipo: Boolean
 - Descripción: Indica si el envío es contra entrega (con recaudo), o no. Enviar **true** cuando sea contra entrega, enviar **false** cuando el vaya con pago anticipado (sin recaudo).
 - Requerido: SI.
 - Ejemplo: true
- amount:
 - tipo: int
 - Descripción: El valor total de la orden.
 - Requerido: SI.
 - Ejemplo: 75000
- ciudad_destino:
 - tipo: Object
 - Descripción: Este parámetro, es un objeto, con el código dane de la ciudad destino. Dentro de este parámetro, debe ir un índice, llamado **cod_dane**, el cual es del tipo string, y debe contener el código dane de la ciudad, el cual contiene 7 caracteres
 - Requerido: SI.
 - Ejemplo:

```
"ciudad_destino":{
  "cod_dane":"5001000"
}
```
- ciudad_remitente:
 - tipo: Object
 - Descripción: Este parámetro, es un objeto, con el código dane de la ciudad del remitente, es decir, desde donde va a salir el producto. Dentro de este parámetro, debe ir un índice, llamado **cod_dane**, el cual es del tipo string, y debe contener el código dane de la ciudad, el cual contiene 7 caracteres
 - Requerido: SI.
 - Ejemplo:

```
"ciudad_remitente":{
  "cod_dane":" 11001000"
}
```

```
}
```

Adjunto ejemplo de la respuesta:

```
{
  "isSuccess": true,
  "status": 200,
  "objects": [
    {
      "distributionCompany": {
        "id": 1,
        "name": "SERVIENTREGA",
        "created_at": "2022-05-14 15:03:05",
        "updated_at": null,
        "deleted_at": null,
        "incidence_solution_method": "MEC2",
        "is_visible": true,
        "last_order_id_cron_update_status": null
      },
      "transportadora": "SERVIENTREGA",
      "transportadora_minuscula": "servientrega",
      "transportadora_id": 1,
      "objects": {
        "precioEnvio": 12050,
        "trayecto": "",
        "overload_was_applied": false
      }
    },
  ],
}
```

```
{
  "distributionCompany": {
    "id": 2,
    "name": "ENVIA",
    "created_at": "2022-05-14 15:03:05",
    "updated_at": "2022-09-26 11:30:21",
    "deleted_at": null,
    "incidence_solution_method": "URL",
    "is_visible": false,
    "last_order_id_cron_update_status": "2497175"
  },
  "transportadora": "ENVIA",
  "transportadora_minuscula": "envia",
  "transportadora_id": 2,
  "objects": {
    "precioEnvio": 12800,
    "trayecto": "NACIONAL",
    "overload_was_applied": false
  }
},
{
  "distributionCompany": {
    "id": 5,
    "name": "COORDINADORA",
    "created_at": "2023-01-16 05:36:59",
    "updated_at": null,
    "deleted_at": null,
    "incidence_solution_method": "URL",
    "is_visible": true,
```

```

    "last_order_id_cron_update_status": null
  },
  "transportadora": "COORDINADORA",
  "transportadora_minuscula": "coordinadora",
  "transportadora_id": 5,
  "objects": {
    "precioEnvio": 15366,
    "trayecto": "",
    "overload_was_applied": false
  }
},
}

```

Adjunto imagen de ejemplo de respuesta:

```

{
  "isSuccess": true,
  "status": 200,
  "objects": [
    {
      "distributionCompany": {
        "id": 1,
        "name": "SERVIENTREGA",
        "created_at": "2022-05-14 15:03:05",
        "updated_at": null,
        "deleted_at": null,
        "incidence_solution_method": "MEC2",
        "is_visible": true,
        "last_order_id_cron_update_status": null
      },
      "transportadora": "SERVIENTREGA",
      "transportadora_minuscula": "servientrega",
      "transportadora_id": 1,
      "objects": {
        "precioEnvio": 12050,
        "trayecto": "",
        "overload_was_applied": false
      }
    },
    {
      "distributionCompany": {
        "id": 2,
        "name": "ENVIA",
        "created_at": "2022-05-14 15:03:05",
        "updated_at": "2022-09-26 11:30:21",
        "deleted_at": null
      }
    }
  ]
}

```

* Productos:

La siguiente lista, es el listado de servicios que se pueden integrar.

- Listar Categorías.
- Crear Producto.
- Listar Productos.
- Listar Producto específico.

A continuación, el api para obtener la lista de Categorías:

- Url de Prueba: <https://test-api.dropi.co/api/categories>
- Url de Producción: <https://api.dropi.co/api/categories>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Adjunto un ejemplo de respuesta exitosa:

Body Cookies Headers (12) Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1  {
2      "isSuccess": true,
3      "count": 2,
4      "status": 200,
5      "objects": [
6          {
7              "id": 1,
8              "name": "HOGAR",
9              "parent_category": 0
10         },
11         {
12             "id": 2,
13             "name": "COCINA",
14             "parent_category": 0
15         }
16     ]
17 }
```

A continuación, el api para la creación de Productos:

- Url de Prueba: <https://test-api.dropi.co/api/products>
- Url de Producción: <https://api.dropi.co/api/products>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros:

- categories:
 - tipo: object
 - Descripción: Categorías a la que va a pertenecer el producto. Se pueden enviar varias categorías, que deben venir del api de categorías.
 - Requerido: SI.
 - Ejemplo: [{"id": 1, "name": "Hogar", "parent_category": 0}]
- name:
 - tipo: string
 - Descripción: Nombre del producto, se puede agregar cualquier texto. Cantidad máxima recomendada 45 caracteres.
 - Requerido: SI
 - Ejemplo: Gancho para pantalones de 15 cm.
- description:
 - tipo: string
 - Descripción: Descripción del producto, se puede agregar cualquier texto.
 - Requerido: SI
 - Ejemplo: Gancho para pantalones de 15 cm.
- private_product:
 - tipo: Booleano
 - Descripción: Indica si el producto es privado para el usuario o es público.
 - Requerido: SI
 - Ejemplo: TRUE o FALSE.
- sale_price:
 - tipo: float
 - Descripción: Costo del producto que le costó importarlo o hacerlo puesto en la bodega.
 - Requerido: SI
 - Ejemplo: 12000.
- suggested_price:
 - tipo: float
 - Descripción: Precio de venta final, del producto.
 - Requerido: SI
- sku:
 - tipo: string

- Descripción: Código de identificación del producto, ingresado manualmente por el usuario. Si no se envía, se asignará uno automáticamente.
- Requerido: NO
- Ejemplo: gancho-ropa-23
- stock:
 - tipo: int
 - Descripción: Indica la cantidad de unidades disponibles del producto, el stock oficial.
 - Requerido: SI
 - Ejemplo: 500
- type:
 - tipo: string
 - Descripción: Indica si el producto es SIMPLE o VARIABLE.
 - Requerido: SI
 - Ejemplo: SIMPLE.
- attribute:
 - tipo: object
 - Descripción: Atributos del producto, solo en caso de que el producto sea VARIABLE, que va a pertenecer el producto. Se pueden enviar varios atributos, puede ir vacío: [].
 - Requerido: SI.
 - Ejemplo:


```
[
    {
      "deleted_at": null,
      "description": "COLOR",
      "id": 0,
      "isVariation": true,
      "product_id": 0,
      "values": [
        {
          "attribute_id": 0,
          "attribute_name": "COLOR",
          "deleted_at": null,
          "id": 0,
          "value": "ROJO"
        },
        {
          "attribute_id": 0,
          "attribute_name": "COLOR",
          "deleted_at": null,
          "id": 0,
          "value": "AMARILLO"
        }
      ]
    }
  ],
```



```

{
  "deleted_at": null,
  "description": "TALLA",
  "id": 0,
  "isVariation": true,
  "product_id": 0,
  "values":[
    {
      "attribute_id": 0,
      "attribute_name": "TALLA",
      "deleted_at": null,
      "id": 0,
      "value": "M"
    },
    {
      "attribute_id": 0,
      "attribute_name": "TALLA",
      "deleted_at": null,
      "id": 0,
      "value": "L"
    }
  ]
}
]

```

Nótese que cada atributo tiene su nombre, y dentro, tiene un parámetro llamado values, que contiene los valores del atributo, ejemplo: atributo TALLA, values M,L,XL etc.

- variations:
 - tipo: object
 - Descripción: Variaciones del producto, en caso de ser tipo VARIABLE. Puede ir vacío si no es variable, ejemplo []
 - Requerido: SI.
 - Ejemplo:

```

[
  {
    "deleted_at": null
    "id": 0
    "product_id": 0
    "sale_price": 12000
    "sku": "sku"
    "stock": "200"
    "suggested_price": 60000,
    "attribute_values":[
      {

```

```

        "attribute_name": "TALLA",
        "deleted_at": null,
        "id": "M",
        "value": "M"
    },
    {
        "attribute_name": "COLOR",
        "deleted_at": null,
        "id": "ROJO",
        "value": "ROJO"
    }
]
}
]

```

Nótese que cada variación, tiene su propio stock, su precio, y su precio sugerido.

Cada variación, también tiene un parámetro llamado “attribute_values”, el cual contiene los atributos que se van a crear para el producto, por ejemplo: TALLA, COLOR, que vienen de la variable llamada “attribute” del producto.

- id:
 - tipo: int
 - Descripción: Este campo debe ir en NULL, cuando se está creando un producto, para que se le asigne un id, solo se tomará en cuenta cuando se ingrese un valor entero, y ese producto pertenezca al usuario, para editarlo.
 - Requerido: SI
 - Ejemplo: NULL, 35.
- weight:
 - tipo: float
 - Descripción: Peso del producto en kg, puede incluir decimales.
 - Requerido: SI
 - Ejemplo: 2.
- width:
 - tipo: float
 - Descripción: Ancho del producto, puede incluir decimales.
 - Requerido: SI
 - Ejemplo: 20.
- length:
 - tipo: float
 - Descripción: Longitud del producto, puede incluir decimales.
 - Requerido: SI
 - Ejemplo: 20.

- height:
 - tipo: float
 - Descripción: Altura del producto, puede incluir decimales.
 - Requerido: SI
 - Ejemplo: 20.
- warehouseselected:
 - tipo: int
 - Descripción: Indica el id de la bodega a la cual se va a asignar el producto. El usuario va a poder agregar productos a su propia bodega que previamente haya creado.
 - Requerido: SI.
 - Ejemplo: 45
- add_stock_in_return:
 - tipo: Booleano
 - Descripción: Indica si al momento de una devolución, la cantidad devuelta se suma al stock o no.
 - Requerido: SI
 - Ejemplo: TRUE o FALSE.

Adjunto arreglo de datos, enviados desde postman:

```
{
  "categories": [
    {
      "id": 1,
      "name": "Hogar",
      "parent_category": 0
    }
  ],
  "description": "Gancho para pantalones de 15 cm",
  "height": 0,
  "id": null,
  "length": 0,
  "name": "Gancho para Pantalones",
  "privated_product": true,
  "sale_price": 10000,
  "sku": "gancho-ropa-23",
  "stock": 300,
  "suggested_price": 60000,
  "type": "SIMPLE",
  "user_profit_percentage_product": [],
  "variations": [
    {
      "deleted_at": null,
      "id": 0,
      "product_id": 0,
      "sale_price": 35,
      "sku": "-0",
      "stock": "200",
      "suggested_price": 500,
      "attribute_values": [
        {

```

```

        "attribute_name": "TALLA",
        "deleted_at": null,
        "id": "M",
        "value": "M"
    },
    {
        "attribute_name": "COLOR",
        "deleted_at": null,
        "id": "ROJO",
        "value": "ROJO"
    }
]
},
"attributes": [
    {
        "deleted_at": null,
        "description": "COLOR",
        "id": 0,
        "isVariation": true,
        "product_id": 0,
        "values": [
            {
                "attribute_id": 0,
                "attribute_name": "COLOR",
                "deleted_at": null,
                "id": 0,
                "value": "ROJO"
            },
            {
                "attribute_id": 0,
                "attribute_name": "COLOR",
                "deleted_at": null,
                "id": 0,
                "value": "AMARILLO"
            }
        ]
    },
    {
        "deleted_at": null,
        "description": "TALLA",
        "id": 0,
        "isVariation": true,
        "product_id": 0,
        "values": [
            {
                "attribute_id": 0,
                "attribute_name": "TALLA",
                "deleted_at": null,
                "id": 0,
                "value": "M"
            },
            {
                "attribute_id": 0,
                "attribute_name": "TALLA",
                "deleted_at": null,
                "id": 0,
                "value": "L"
            }
        ]
    }
]

```

```

    ]
  }
],
"weight":1,
"width":1
}

```

Adjunto un ejemplo de respuesta exitosa:

```

{
  "isSuccess": true,
  "message": "El item ha sido creado con exito!.",
  "status": 200,
  "objects": [
    {
      "name": "Gancho para Pantalones",
      "description": "Gancho para pantalones de 15 cm",
      "type": "SIMPLE",
      "stock": 300,
      "sale_price": 10000,
      "suggested_price": 60000,
      "user_id": 12,
      "privated_product": true,
      "active": true,
      "sku": "gancho-ropa-23",
      "weight": 1,
      "length": 0,
      "width": 1,
      "height": 0,
      "dropi_percent_product_supplier": null,
      "dropi_percent_product_drops": null,
      "updated_at": "2022-08-16 17:54:31",
      "created_at": "2022-08-16 17:54:31",
      "id": 6,
      "is_warehouse_dropi": false
    }
  ],
  "notification": ""
}

```

A continuación, el api para obtener los productos:

- Url de Prueba: <https://test-api.dropi.co/api/products/index>
- Url de Producción: <https://api.dropi.co/api/products/index>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros:

- keywords:
 - tipo: string
 - Descripción: Texto a buscar, que coincida con el nombre, descripción, sku del producto a buscar. Puede ir vacío "".
 - Requerido: NO
- pageSize:
 - tipo: int
 - Descripción: Cantidad de resultados que se quiere obtener.
 - Requerido: NO
 - Ejemplo: 50
- startData:
 - tipo: int
 - Descripción: Indica la fila en la que se quiere obtener los resultados.
 - Requerido: NO
 - Ejemplo: 0, El 0 indica que traerá los datos desde el primer producto.
- userVerified:
 - tipo: Boolean
 - Descripción: Si se quiere que se traigan todos los productos, no debe enviarse este parámetro. Si se envía en **true**, retorna los productos en el cual, el usuario haya sido verificado. Si se coloca en **false**, traerá los productos en el cual, el usuario no haya sido verificado.
 - Requerido: NO
 - Ejemplo: true

Adjunto un ejemplo de la solicitud en Postman:

```
{  
  "keywords": "",  
  "pageSize": 10,  
  "startData": 0,  
}
```

Adjunto un ejemplo de respuesta exitosa:

```
1  {
2    "isSuccess": true,
3    "count": 8,
4    "status": 200,
5    "objects": [
6      {
7        "id": 14,
8        "id_lista": 0,
9        "name": "Gancho para Pantalones",
10       "name_in_order": "Gancho para Pantalones - $60000.00 - Stock 300.00",
11       "description": "Gancho para pantalones de 15 cm",
12       "type": "SIMPLE",
13       "stock": "300.00",
14       "sale_price": "10000.00",
15       "suggested_price": "60000.00",
16       "user_id": 12,
17       "privated_product": true,
18       "active": true,
19       "sku": "gancho-ropa-23",
20       "weight": "1.00",
21       "length": "0.00",
22       "width": "1.00",
23       "height": "0.00",
24       "deleted_at": null,
25       "created_at": "2022-08-16T23:55:54.000000Z"
```

A continuación, el api para obtener la información de un Producto específico:

- Url de Prueba: <https://test-api.dropi.co/api/products/IDPRODUCTO>
- Url de Producción: <https://api.dropi.co/api/products/IDPRODUCTO>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Adjunto un ejemplo de respuesta exitosa:

```
1  {
2    "isSuccess": true,
3    "status": 200,
4    "objects": {
5      "id": 14,
6      "id_lista": 0,
7      "name": "Gancho para Pantalones",
8      "name_in_order": "Gancho para Pantalones - $60000.00 - Stock 300.00",
9      "description": "Gancho para pantalones de 15 cm",
10     "type": "SIMPLE",
11     "stock": "300.00",
12     "sale_price": "10000.00",
13     "suggested_price": "60000.00",
14     "user_id": 12,
15     "privated_product": true,
16     "active": true,
17     "sku": "gancho-ropa-23",
18     "weight": "1.00",
19     "length": "0.00",
20     "width": "1.00",
21     "height": "0.00",
22     "deleted_at": null,
23     "created_at": "2022-08-16T23:55:51.000000Z",
24     "updated_at": null,
25     "is_warehouse_dropi": false,
26     "dropi_percent_product_supplier": null,
27     "dropi_percent_product_drops": null,
```


* Novedades:

La siguiente lista, es el listado de servicios que se pueden integrar.

- Listar Novedades pendientes por solucionar.

A continuación, el api para Listar Novedades Pendientes por Solucionar:

- Url de Prueba: <https://test-api.dropi.co/ap/orders/myorders>
- Url de Producción: <https://api.dropi.co/api/orders/myorders>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros:

- result_number:
 - tipo: int
 - Descripción: Cantidad de resultados (órdenes) que quieres que te devuelva el api.
 - Requerido: NO (pero es recomendable hacerlo).
 - Ejemplo: 10
- start:
 - tipo: int
 - Descripción: Descripción: Valor inicial desde el cual se quiere obtener los resultados.
 - Requerido: NO.
 - Ejemplo: 1
- textToSearch:
 - tipo: string
 - Descripción: Texto, el cual buscará si coincide con: estatus, nombre de cliente, apellidos del cliente, dirección, departamento, ciudad, nombre de producto.
 - Requerido: NO.
- from_date_last_incidence:
 - tipo: string
 - Descripción: Filtro para filtrar por fecha de la novedad reportada más reciente, este campo es "Desde" la fecha en la que se quiere filtrar. El formato es: yyyy-mm-dd
 - Requerido: NO.
 - Ejemplo: 2022-10-07
- until_date_last_incidence:
 - tipo: string

- Descripción: Filtro para filtrar por fecha de la novedad reportada más reciente, este campo es “Hasta” la fecha en la que se quiere filtrar. El formato es: yyyy-mm-dd
 - Requerido: NO.
 - Ejemplo: 2022-11-07
- status:
 - tipo: string
 - Descripción: Para filtrar por estatus.
 - Requerido: NO.
 - Ejemplo: PENDIENTE, GUIA_GENERADA
- haveIncidenceProcesamiento:
 - tipo: Boolean
 - Descripción: Enviar en true para recibir las órdenes que están pendiente por solucionar, que aun no hayan sido gestionadas, ni por parte del usuario, ni por parte de la transportadora.
 - Requerido: SI.
 - Ejemplo: true
- issue_solved_by_parent_order:
 - tipo: Boolean
 - Descripción: Enviar en false para recibir las órdenes que están pendiente por solucionar, que aún no hayan sido gestionadas por el usuario.
 - Requerido: SI.
 - Ejemplo: false

* TICKETS:

La siguiente lista, es el listado de servicios que se pueden integrar.

- Crear un ticket.
- Crear mensajes.
- Listar conversación de un ticket.
- Listar todos mis tickets

A continuación, el api para crear un ticket específico:

- Url de Prueba: <https://test-api.dropi.co/api/tickets/createticket>
- Url de Producción: <https://api.dropi.co/api/tickets/createticket>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Por URL:

- Url de Prueba:
https://test-api.dropi.co/api/tickets/createticket?ticket_name=prueba&phone=36232821&filtro=ORDENID&order_id=84&description=funciona perfectamente
- Url de Producción:
https://api.dropi.co/api/tickets/createticket?ticket_name=prueba&phone=36232821&filtro=ORDENID&order_id=84&description=funciona perfectamente
- Método: GET
- Authorization: Bearer TOKEN

Parámetros:

- ticket_name:
 - tipo: string
 - Descripción: . va el nombre del ticket.
 - Requerido: SI
 - Ejemplo: PRUEBA TICKET1
- phone:
 - tipo: string
 - Descripción: seria el telefono del cual se va atender el ticket, solo se permiten numeros.
 - Requerido: SI
 - Ejemplo: 3135654489
- order_id:
 - tipo: int
 - Descripción: Indica a cuál orden id o guia que pertenece el ticket, (debe ser una orden que exista actualmente).
 - Requerido: SI

- Ejemplo: 84.
- filtro:
 - tipo: string
 - Descripción: solo se le envía como parámetro 'GUIA' o 'ORDEN ID'.
 - Requerido: SI
 - Ejemplo: "ORDEN ID"
- description:
 - tipo: string
 - Descripción: sería la descripción de la conversación de un ticket.
 - Requerido: SI
 - Ejemplo: "me gustaría tener una solución al problema x que se me presento"

Adjunto un ejemplo de la solicitud en Postman:

```
{
  "ticket_name": "PRUEBA TICKET",
  "phone": "3135654489",
  "order_id": 84,
  "filtro": "ORDEN ID",
  "description": "me gustaría tener una solución al problema x que se me presento"
}
```

Adjunto un ejemplo de respuesta exitosa:

```
{
  "isSuccess": true,
  "status": 200,
  "objects": {
    "ticket_name": "PRUEBA TICKET1",
    "status": "POR ASIGNAR",
    "phone": "3135654489",
    "order_id": 84,
    "create_by": 2,
    "user_id": 2,
    "topic_ticket_id": null,
    "assigned_to_id": null,
    "updated_at": "2022-10-20 12:22:04",
    "created_at": "2022-10-20 12:22:04",
    "id": 198
  },
}
```

A continuación, el api para crear una conversación de un ticket específico (si desean crear mensajes dentro de un ticket):

- Url de Prueba: <https://test-api.dropi.co/api/tickets/createconversationticket>
- Url de Producción: <https://api.dropi.co/api/tickets/createconversationticket>
- Método: POST
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

Por URL:

- Url de Prueba:
https://test-api.dropi.co/api/tickets/createconversationticket?ticket_id=192&description=respondo a esta conversacion
- Url de Producción:
https://api.dropi.co/api/tickets/createconversationticket?ticket_id=192&description=respondo a esta conversacion
- Método: GET
- Authorization: Bearer **TOKEN**

Parámetros:

- ticket_id:
 - tipo: int
 - Descripción: aquí iría el id del ticket creado anteriormente.
 - Requerido: SI
 - Ejemplo: 192
- description:
 - tipo: string
 - Descripción: sería la descripción de la conversación de un ticket.
 - Requerido: SI
 - Ejemplo: "me gustaría tener una solución al problema x que se me presento"

Adjunto un ejemplo de la solicitud en Postman:

```
{  
  "ticket_id":192,  
  "description":"me gustaría tener una solución al problema x que se me presento"  
}
```

Adjunto un ejemplo de respuesta exitosa:

```
{
  "isSuccess": true,
  "status": 200,
  "objects": {
    "user_id": 2,
    "ticket_id": 192,
    "description": "me gustaría tener una solución al problema x que se me presento",
    "updated_at": "2022-10-19 17:02:50",
    "created_at": "2022-10-19 17:02:50",
    "id": 253
  },
}
```

A continuación, el api para ver una conversación de un ticket específico:

- Url de Prueba: <https://test-api.dropi.co/api/tickets/indexconversation>
- Url de Producción: <https://api.dropi.co/api/tickets/indexconversation>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer **TOKEN**

Por URL:

- Url de Prueba: <https://test-api.dropi.co/api/tickets/indexconversation?idSelect=192>
- Url de Producción: <https://api.dropi.co/api/tickets/indexconversation?idSelect=192>
- Método: GET
- Authorization: Bearer **TOKEN**

Parámetros:

- idSelect:
 - tipo: int
 - Descripción: aquí iría el id del ticket creado anteriormente para poder visualizarlo.
 - Requerido: SI
 - Ejemplo: 192

Adjunto un ejemplo de la solicitud en Postman:

```
{
  "idSelect":192
}
```

Adjunto un ejemplo de respuesta exitosa:

```
{
  "isSuccess": true,
  "object": [
    {
      "id": 253,
      "user_id": 2,
      "ticket_id": 192,
      "description": "me gustaría tener una solución al problema x que se me presento",
      "created_at": "2022-10-19 17:02:50",
      "updated_at": "2022-10-19 17:02:50",
      "deleted_at": null,
      "user": {
        "id": 2,
        "name": "Francisco",
        "email": "franciscoramirez@gmail.com",
        "password": "12345678"
      }
    }
  ]
}
```

A continuación, el api para ver los listados de mis tickets:

- Url de Prueba: <https://test-api.dropi.co/api/tickets/indextickets>
- Url de Producción: <https://api.dropi.co/api/tickets/indextickets>
- Método: GET
- Cabeceras: Content-Type: application/json, Authorization: Bearer TOKEN

Parámetros para filtrar:

- status:
 - tipo: string
 - Descripción: para poder visualizar por qué estado están tus tickets.
 - Requerido: NO
 - Ejemplo: 'POR ASIGNAR', 'ASIGNADO', 'EN PROCESO', 'RESUELTO'
 - EjemploUrl de Prueba: <https://test-api.dropi.co/api/tickets/indextickets?status=POR ASIGNAR>
 - EjemploUrl de Producción: <https://api.dropi.co/api/tickets/indextickets?status=POR ASIGNAR>

Adjunto un ejemplo de la solicitud en Postman:

```
{
  "status": "POR ASIGNAR"
}
```

- search:
 - tipo: string
 - Descripción: para poder buscar tickets por el nombre.
 - Requerido: NO

- Ejemplo: 'ticket prueba'
- EjemploUrl de Prueba: <https://test-api.dropi.co/api/tickets/indextickets?search=PRUEBA TICKET>
- EjemploUrl de Producción: <https://api.dropi.co/api/tickets/indextickets?search=PRUEBA TICKET>

Adjunto un ejemplo de la solicitud en Postman:

```
{
  "search": "PRUEBA TICKET"
}
```

- filterBy:
 - tipo: string
 - Descripción: para poder filtrar por el id del ticket, debe buscar con el search y este parámetro.
 - Requerido: NO
 - Ejemplo: 'TICKET ID'
 - EjemploUrl de Prueba: <https://test-api.dropi.co/api/tickets/indextickets?search=171&filterBy=TICKET ID>
 - EjemploUrl de Producción: <https://api.dropi.co/api/tickets/indextickets?search=171&filterBy=TICKET ID>

Adjunto un ejemplo de la solicitud en Postman:

```
{
  "search": 171,
  "filterBy": "TICKET ID"
}
```

- notShowResuelto:
 - tipo: booleano
 - Descripción: por si quieres ver los tickets resueltos o no, a la hora de listar todos los tickets, true es para no ver los tickets resueltos y false es para verlos.
 - Requerido: NO
 - Ejemplo: true or false
 - EjemploUrl de Prueba: <https://test-api.dropi.co/api/tickets/indextickets?notShowResuelto=false>
 - EjemploUrl de Producción: <https://api.dropi.co/api/tickets/indextickets?notShowResuelto=false>

```
{
  "notShowResuelto": false
}
```


Adjunto un ejemplo de respuesta exitosa:

```
1  {
2    "isSuccess": true,
3    "object": [
4      {
5        "id": 154,
6        "ticket_name": "no resuelto",
7        "status": "RESUELTO",
8        "phone": "3104167805",
9        "order_id": null,
10       "create_by": 13,
11       "user_id": 2,
12       "created_at": "2022-08-25 16:32:36",
13       "updated_at": "2022-08-26 10:45:51",
14       "deleted_at": null,
15       "topic_ticket_id": 15,
16       "assigned_to_id": 9,
```