

API Custom | Batch Pedidos

- [O que é o API Custom | Pedidos](#)
 - [Construindo o CSV](#)
 - [Estrutura do CSV](#)
 - [Dados e Validação](#)
 - [Acessando a API](#)
 - [Importando os dados](#)
 - [Payload da requisição](#)
 - [Payload de retorno](#)
 - [Created - 201](#)
 - [Bad Request - 400](#)
 - [Unauthorized - 401](#)
 - [Unprocessable Entity - 422](#)
 - [INTERNAL SERVER ERROR - 500](#)
 - [Listando as importações realizadas](#)
 - [Payload de requisição](#)
 - [Payload de retorno](#)
 - [Success - 200](#)
 - [Unauthorized - 401](#)
 - [Not Found - 404](#)

O que é o API Custom | Pedidos

A **API Custom | Batch Pedidos** permite importação em lotes de dados relacionados a Pedidos originados em plataformas ou bases externas ao ZENVIA Customer Cloud. Os Dados importados auxiliam em funcionalidades da plataforma como o **Cálculo de RFV**, onde são utilizados em conjunto aos dados importados pelas APIs listadas abaixo:

- [API Custom | Batch Produtos](#)
- [API Custom | Batch Notas Fiscais](#)
- [API Custom | Batch Contatos](#)

Construindo o CSV

Para utilizar a API de Pedidos é necessário montar um CSV com base em uma estrutura pré-definida e em conformidade com os tipos de dados e suas respectivas validações. Abaixo, você pode ler detalhadamente sobre este padrão:

Estrutura do CSV

O CSV deve seguir uma estrutura simples onde:

- A primeira linha deve ser o cabeçalho da tabela;
- As colunas devem estar em uma ordem pré-definida;
- O CSV deve separar as colunas por ponto e vírgula `;`.

A ordem pré-definida deve ser a seguinte:

1. `externalPlatform`
2. `externalId`
3. `invoiceExternalId`
4. `customerExternalId`
5. `orderNumber`
6. `orderTimestamp`
7. `totalAmount`
8. `legalCode`
9. `legalName`
10. `primaryPhone`
11. `email`
12. `orderStatus`
13. `productExternalId`
14. `sku`
15. `ean`
16. `name`
17. `quantity`
18. `measurementUnit`
19. `priceUnit`

Dados e Validação

Os dados da tabela são organizados da seguinte forma:

Campo	Tipo	Obrigatório	Descrição
<code>externalPlatform</code>	<code>string</code>		Identificação do ERP de origem.

externalId	string	✓	Código interno do sistema de origem.
invoiceExternalId	string	✗	Código interno da nota no sistema de origem
customerExternalId	string	✗	Código interno de <i>customer</i> (contato) no sistema de origem
orderId	string	✓	Número de identificação do pedido.
orderTimestamp	string	✓	Data e hora do pedido.
totalAmount	number	✓	Valor total do pedido.
legalCode	string	✗	Documento legal do destinatário (CPF, CNPJ, etc.)
legalName	string	✗	Uso para clientes não cadastrados.
primaryPhone	string	✗	Número de telefone principal do cliente.
email	string	✗	Endereço de e-mail do cliente.
orderStatus	string	✓	Status atual do pedido.

<code>productExternalId</code>	<code>string string[]</code>	✗	Código interno do produto no sistema de origem.
<code>sku</code>	<code>string string[]</code>	✗	Código SKU do produto.
<code>ean</code>	<code>string string[]</code>	✗	Código de barras do produto.
<code>name</code>	<code>string string[]</code>	✓	Nome do produto.
<code>quantity</code>	<code>string string[]</code>	✓	Quantidade solicitada do produto.
<code>measurementUnit</code>	<code>string string[]</code>	✗	Unidade de medida do produto.
<code>priceUnit</code>	<code>string string[] number number[]</code>	✓	Preço unitário do produto.

Para estes dados, são realizadas as seguintes validações:

Campos que aceitam listas como `quantity`, `measurementUnit` e `priceUnit` devem usar a vírgula “,” e espaço como separador, exemplo:

- `400.5, 500,00, 356.56`

O campo `externalPlatform` aceita apenas os valores do seguinte `enum` :

```
1 ['OMIE', 'BLING', 'TINY', 'MICROVIX', 'WBUY', 'OTHER'];
```

Obs.: Os valores podem ser inseridos com letras minúsculas, exemplo: `omie`

O campo `email` tem as seguintes validações:

- Deve começar com letra ou número

- Não pode possuir acentuação
- Não pode ter caracteres especiais
 - Com exceção de `"_"`, `"."`, `"@"`, `"-"`
- Deve conter `"@"`

O campo `primaryPhone` tem as seguintes validações:

- DDD é obrigatório
- DDD deve ser válido no Brasil
- Aceita entre 8 e 9 caracteres (considerando apenas o número sem DDD e DDI)
- Não aceita caracteres especiais
 - Com exceção de `"(""`, `"-"`
- Letras não são aceitas

Lista de DDDs válidos no Brasil:

```
1 11, 12, 13, 14, 15, 16, 17, 18, 19, 21,
2 22, 24, 27, 28, 31, 32, 33, 34, 35, 37,
3 38, 41, 42, 43, 44, 45, 46, 47, 48, 49,
4 51, 53, 54, 55, 51, 61, 62, 63, 64, 65,
5 66, 67, 68, 69, 71, 73, 74, 75, 77, 79,
6 81, 82, 83, 84, 85, 86, 87, 88, 89, 91,
7 92, 93, 94, 95, 96, 97, 98, 99.
```

O campo `orderStatus` aceita apenas os seguintes valores:

```
1 [
2   'WAITING_PAYMENT',
3   'PAYMENT_ANALISYS',
4   'PAYMENT_CONFIRMED',
5   'ORDER_EXPEDITING',
6   'ORDER_SHIPPED',
7   'ORDER_CANCELED',
8   'ORDER_FINISHED',
9   'PAYMENT_DENIED',
10  'ORDER_AVAILABLE',
11  'ORDER_RETURNED',
12  'ORDER_REVIEW',
13  'INVOICE_ISSUED',
14  'ORDER_PREPARING',
15  'RECEIPT_ISSUED',
16  'ON_ROUTE',
17  'PICKUP_AVAILABLE',
18  'DELIVERED',
19 ];
```

Obs.: Os valores podem ser inseridos com letras minúsculas, exemplo: `delivered`

O campo `measurementUnit` aceita apenas os seguintes valores:

```
1  [  
2  'AMPOLA',  
3  'BALDE',  
4  'BANDEJ',  
5  'BARRA',  
6  'BISNAG',  
7  'BLOCO',  
8  'BOBINA',  
9  'BOMB',  
10 'CAPS',  
11 'CART',  
12 'CENTO',  
13 'CJ',  
14 'CM',  
15 'CM2',  
16 'CX',  
17 'CX2',  
18 'CX3',  
19 'CX5',  
20 'CX10',  
21 'CX15',  
22 'CX20',  
23 'CX25',  
24 'CX50',  
25 'CX100',  
26 'DISP',  
27 'DUZIA',  
28 'EMBAL',  
29 'FARDO',  
30 'FOLHA',  
31 'FRASCO',  
32 'GALAO',  
33 'GF',  
34 'GRAMAS',  
35 'JOGO',  
36 'KG',  
37 'KIT',  
38 'LATA',  
39 'LITRO',  
40 'M',  
41 'M2',  
42 'M3',  
43 'MILHEI',  
44 'ML',  
45 'MWH',  
46 'PACOTE',  
47 'PALETE',  
48 'PARES',  
49 'PC',  
50 'POTE',  
51 'K',  
52 'RESMA',  
53 'ROLO',  
54 'SACO',  
55 'SACOLA',  
56 'TAMBOR',  
57 'TANQUE',  
58 'TON',
```

```
59 'TUBO',
60 'UNID',
61 'VASIL',
62 'VIDRO',
63 ];
```

Obs.: Os valores podem ser inseridos com letras minúsculas, exemplo: **ampola**

Acessando a API

Para acessar e consumir a API você pode utilizar o software que desejar, como *Postman* ou *Insomnia*, ou até mesmo criar uma interface de usuário (UI) para consumi-la. Porém é necessário possuir um token válido gerado na interface do ZENVIA Customer Cloud.

A geração do token pode ser realizada no seguinte endereço: <https://app.zenvia.com/settings/api>.

Importando os dados

Para importar os pedidos pela API, é necessário realizar uma requisição **POST** na seguinte rota da API:

<https://api.zenvia.com/v2/order-batches>

Payload da requisição

Payload utilizado para realizar a importação dos pedidos:

```
1 curl -s \
2 --header "X-API-TOKEN: <SEU_API_TOKEN>" \
3 --header "Content-Type: multipart/form-data" \
4 --form "file=<SEU_CSV>" \
5 --location "https://api.zenvia.com/v2/order-batches"
```

Payload de retorno

Payloads que a API retorna ao tentar realizar uma importação:

Created - 201

Validação com sucesso

```
1 {
2   "id": "dc514337-14c4-4f53-947b-93088b450e2e",
3   "organizationId": "a49c0383-57b1-4627-bfaa-6019b7e7fe53",
4   fileName: "Pedidos - Lote 1",
5   size: "24 Mb",
6   status: "SUCCESS",
7   createdAt: "2025-04-23T18:42:50.325Z"
8 }
```

Validação com erro:

```
1 {
2   "id": "a311f0dd-4be7-4cec-a6b0-65c13c3c83ac",
3   "organizationId": "a49c0383-57b1-4627-bfaa-6019b7e7fe53",
4   fileName: "Pedidos - Lote 1",
5   size: "24 Mb",
```

```
6   status: "FAILED",
7   createdAt: "2025-04-23T18:42:50.325Z"
8 }
```

⚠ Esta mensagem ocorre apenas quando **todos os dados do arquivo são inválidos**.

Bad Request - 400

```
1 {
2   "message": "<MENSAGEM_DE_ERRO>",
3   "error": "Bad Request",
4   "statusCode": 400,
5 }
```

Pode ocorrer em 2 casos:

- Arquivo não é enviado: **File is Required**
- Arquivo é enviado na chave errada: **Unexpected field - <KEY_INCORRETA>**

Unauthorized - 401

```
1 {
2   "code": "AUTHENTICATION_ERROR",
3   "message": "Invalid token"
4 }
```

Unprocessable Entity - 422

```
1 {
2   "code": "VALIDATION_ERROR",
3   "message": "Order batch processing error",
4   "details": {
5     code: 422,
6     message: "<MENSAGEM_DE_ERRO>"
7   }
8 }
```

Estes são os possíveis erros e suas mensagens:

- CSV não possui cabeçalho e ao menos 1 linha de dados: **The file must contain at least a header and one data line**
- Nome de coluna incorreta ou fora de ordem: **Header column name '<COLUNA_INVALIDA>' is not correct or out of order**
- Delimitador de colunas inválido: **Invalid delimiter used in the file. Preferably use semicolon (;) as the delimiter.**
- Número de colunas maior ou menor do que o esperado: **Invalid number of columns in the header file**

INTERNAL SERVER ERROR - 500

```
1 {
2   "statusCode": 500,
3   "message": "Internal server error"
4 }
```

Listando as importações realizadas

Para listar as importações de pedidos pela API, é necessário realizar uma requisição **GET** na seguinte rota da API:

<https://api.zenvia.com/v2/order-batches>

Payload de requisição

Payload utilizado para realizar a listagem das importações de pedidos:

```
1 curl -s \
2 --header "X-API-TOKEN: <SEU_API_TOKEN>" \
3 --location "https://api.zenvia.com/v2/order-batches"
```

Payload de retorno

Payloads que a API retorna ao tentar realizar a listagem:

Success - 200

```
1 [
2   {
3     "id": "2565909b-ccd5-4da1-b781-7000718ec65d",
4     "organizationId": "a49c0383-57b1-4627-bfaa-6019b7e7fe53",
5     "fileName": "Pedidos - Lote 1",
6     "lines": 100,
7     "status": "SUCCESS",
8     "consolidatedStatus": {
9       "extractedRecords": 100,
10      "mappedRecords": 100,
11      "successProcessedRecords": 88,
12      "errorProcessedRecords": 12,
13    },
14     "createdAt": "2025-03-18T14:23:03.814Z",
15     "updatedAt": "2025-03-18T14:23:03.814Z"
16   }
17 ]
```

Unauthorized - 401

```
1 {
2   "code": "AUTHENTICATION_ERROR",
3   "message": "Invalid token"
4 }
```

Not Found - 404

```
1 {
2   "code": "NOT_FOUND",
```

```
3 "message": "Path \"v2/order-batchess\" was not found. It has a parameter with a invalid value
4 }
```

Autor	Versão	Data	Observação
@Isaac Hermel Reginato	V.0	30/04/2025	Confecção de documento técnico de utilização da API