

DOCUMENTAÇÃO

API (Application Programming Interfacer)

ver. 3.6

ENTREGANDO
ONLINE

Índice

1. Informações gerais sobre o serviço

- 1.1 Contexto
- 1.2 Objetivo
- 1.3 Nome
- 1.4 Gestor do serviço
- 1.6 Responsável técnico
- 1.7 Requisitos e orientações para o acesso

2. Informações detalhadas sobre o serviço

- 2.1 Metodologia REST
- 2.2 Requisitos de Acesso
- 2.3 Chamada aos serviços
- 2.4 Serviços disponíveis
- 2.5 Consulta Individual de Nota fiscal
 - 2.5.1 Parâmetros de Entrada
 - 2.5.2 Parâmetros de Saída
- 2.6 Consulta Geral por data
 - 2.6.1 Parâmetros de Entrada
 - 2.6.2 Parâmetros de Saída
- 2.7 Inclusão de movimento
 - 2.7.1 Parâmetros de Entrada
 - 2.7.2 Parâmetros de Saída
 - 2.7.3 Exemplo de JSON
- 2.8 Padrão de retorno

1. Informações gerais sobre o serviço

1.1 Contexto

A API do sistema Entregando Online foi desenvolvida com um propósito de integrar os dados cadastrados e consultar o status de notas fiscais.

1.2 Objetivo

O objetivo principal é cadastrar, editar e disponibilizar as informações dos movimentos diários da transportadora no Sistema Entregando Online.

1.3 Nome

API do Entregando Online

1.4 Gestor do Serviço

Nome: Lucas Dos Reis Pereira

E-mail: lucas.pereira@kreativebrasil.com.br

1.6 Responsável técnico

Nome: Lucas Dos Reis Pereira

E-mail: lucas.pereira@kreativebrasil.com.br

1.7 Requisitos e orientações para o acesso

O acesso a API do sistema Entregando Online é restrito, podendo ser acessado somente com informações específicas disponibilizadas pelo Gestor do Serviço descrito acima.

2. Informações gerais sobre o serviço

2.1 Metodologia REST

Representational State Transfer ou somente REST, baseado no design do protocolo HTTP, possui diversos mecanismos embutidos para representar recursos como código de status, representação de tipos de conteúdo, cabeçalhos, etc. O principal nesta arquitetura são as URLs do sistema e os resources (resource é um recurso, entidade). Ele aproveita os métodos HTTP para se comunicar, que são:

GET: Solicita algum recurso como um arquivo ou um script CGI (qualquer dado que estiver identificado pelo URI) por meio do protocolo HTTP. O método GET é reconhecido por todos os servidores.

HEAD: É o mesmo que GET, mas sem que o recurso seja retornado. É usado para obter meta-informações por meio do cabeçalho da resposta, sem ter que recuperar todo o conteúdo.

POST: Envia dados para serem processados (por exemplo, dados de um formulário HTML) para o recurso especificado. Os dados são incluídos no corpo do comando.

DELETE: Remove um recurso.

PUT: Atualiza um recurso na URI especificada. Caso o recurso não exista, ele pode criar um. A principal diferença entre POST e PUT é que o primeiro pode lidar não somente com recursos, mas também pode fazer processamento de informações.

***O Protocolo aceito para esse serviço é somente POST**

2.2 Requisitos de Acesso

Para utilização da API é necessário um parâmetro de entrada chamado IA (Identificação de Acesso) que será fornecido pelo gestor do serviço descrito acima.

2.3 Chamada aos serviços

Todos os serviços utilizados da API devem ser chamados através do seguinte endereço:

<http://global.entregandoonline.com.br/ws/getaccess>

2.4 Serviços disponíveis

Abaixo a lista de serviços disponíveis que a API do Entregando Online oferece.

Para todas as requisições dos serviços os campos (POST) obrigatórios são:

Parâmetro	Tipo	Ocorrência mínima	Observações
IA	int	1	Identificação de acesso
servico	string	1	Nome do serviço
data	string	1	Corpo da mensagem no formato JSON

1º Consulta individual de nota fiscal

Esse tipo de consulta tem como objetivo retornar a situação atual da nota fiscal solicitada pelos parâmetros de entrada da consulta individual de nota

2º Consulta geral por período

Esse tipo de consulta tem como objetivo retornar um grupo de notas fiscais, determinados pelos parâmetros de entrada da consulta geral por período

3º Inclusão de movimento

Esse tipo de serviço tem como objetivo cadastrar e atualizar o movimento do dia determinado pelos parâmetros de entrada pra o serviço inclusão de movimento

2.5 Consulta Individual de Nota fiscal

2.5.1 Parâmetros de Entrada

Parâmetro		Tipo	Ocorrência mínima	Observações	Obrigatório
IA		int	1		Sim
servico		string	1	Deve ser: consulta_nota_fiscal	Sim
data		string	1	Campos em formato JSON	Sim
data (JSON)	NotaFiscal	int	1		Sim
	CTe	int	1		Sim

2.5.2 Parâmetros de Saída

Exemplo de resposta (JSON)

```
[{
  "success": "true",
  "data": {
    "notaFiscal": 1234,
    "cte": 4321,
    "documento": 2341,
    "tipoMov": "ENT",
    "status": "entregue",
    "dataChegada": "2018-02-20 14:30:26",
    "dataAlteracao": "2018-02-20 15:34:10",
    "ocorrencia": "entrega realizada normalmente",
    "ocorrencia_data": "",
    "chaveNfe": "33180500634453000846570000003271961004090119",
    "tempo": "01:04",
    "imagem": "http://entregandoonline.com.br/images/global/ocor435472.jpg"
  }
}]
```

2.6 Consulta Geral por data

2.6.1 Parâmetros de Entrada

Parâmetro		Tipo	Ocorrência mínima	Observações	Obrigatório
IA		int	1		Sim
servico		string	1	Deve ser: consulta_nota_fiscal_per_idodo	Sim
CNPJCliente		string	1	* Esse campo só é usado para clientes da transportadora com acesso específico e token (IA) fornecido separadamente	Não
CNPJCliente (JSON)	cnpj	string	14	CNPJ do cliente da transportadora, apenas números, sem traços, barras e pontos.	Sim
data		string	1	Campos em formato JSON	Sim
data (JSON)	dataEmissao	date	AAAA-MM-DD	Data da criação do documento, (formato 2018-02-20)	Sim
	tempoRetroativo	int	1	Filtra as notas por um período de tempo em minutos	Não
	processado	string	3	Esse parâmetro solicita a exibição das notas que já foram consultadas ou não. Valores permitidos: true , false ou all	Não

Obs.: **tempoRetroativo**

Esse campo tem como objetivo limitar o resultado do grupo de notas que será exibido. Se a API receber esse parâmetro, por exemplo, com o valor 30 serão exibidas as notas com data de alteração dos últimos 30 minutos.

Obs.: **processado**

Quando a chamada ao serviço de consulta_nota_fiscal_periodo é realizada e o parâmetro “*processado*” não é informado, os registros ficam marcados como “lidos” e não serão mais exibidos na consulta. Se desejar consultar novamente esses registros basta passar o parâmetro “*processado*” com o valor **true** (mostrando somente o que já foi “lido”) ou como **all** (exibindo as duas situações, o que foi lido e o que não foi).

2.6.2 Parâmetros de Saída

Exemplo de resposta (JSON)

```
[{
  "success": "true",
  "qtd_documentos": 2,
  "data": [{
    "notaFiscal": 1234,
    "cte": 4321,
    "documento": 2341,
    "tipoMov": "ENT",
    "status": "entregue",
    "dataChegada": "2018-02-20 14:30:26",
    "dataAlteracao": "2018-02-20 15:34:10",
    "ocorrencia": "entrega realizada normalmente",
    "ocorrencia_data": "",
    "chaveNfe": "33180500634453000846570000003271961004090119",
    "tempo": "01:04",
    "imagem": "http://entregandoonline.com.br/images/tsv/ocor435472.jpg",
    "recebedor_data": "",
    "tempo": "0h 1m"
  }, {
    "notaFiscal": 1235,
    "cte": 4325,
    "documento": 2341,
    "tipoMov": "COL",
    "status": "coletado",
    "dataChegada": "2018-02-20 16:30:26",
    "dataAlteracao": "2018-02-20 17:34:10",
    "ocorrencia": "",
    "ocorrencia_data": "",
    "chaveNfe": "33180500634453000846570000003271961004090119",
    "imagem": "http://entregandoonline.com.br/images/tsv/ocor435472.jpg",
    "recebedor_data": "",
    "tempo": "0h 1m"
  }
}]
}]
```


2.7 Inclusão de movimento

2.7.1 Parâmetros de Entrada

Parâmetro		Tipo	Ocorrência mínima	Observações	Obrigatório
IA		int	1		Sim
servico		string	1	Deve ser: importacao	Sim
data		string	1	Campos em formato JSON	Sim
data (JSON)	romaneio	int	1	Número do romaneio	Sim
	codmot	string	14	CPF do motorista	Sim
	nome_mot	string	1	Nome do motorista	Sim
	placa_veic	string	7	Placa do veículo no formato: XPT-3344	Sim
	tel_mot	string	1		Não
	cte	int	1		Sim
	seq_ent	int	1	Número que determina a ordem da lista de execução do movimento	Sim
	nf	int	1		Sim
	volume	int	1		Não
	peso	int	1		Não
	valor_da_mercadoria	double	0.0		Não
	cliente	string	1		Não
	email_cliente	string	1		Não
	nome_do_destino	string	1		Não
	tel_destino	string	1		Não
	cidade_de_destino	string	1		Não
	tipomov	string	3	<u>COL</u> (para coletas) <u>ENT</u> (para entregas)	Sim
	datatransf	date	AAAA-MM-DD	Data em que o movimento será realizado	Sim
	prazoentrega	date	AAAA-MM-DD		Não
	bairro	string	1		Não
	endereco	string	1		Não
	filial	string	3	Sigla da filial, sem espaços e sem caracteres especiais	Sim
	filial_doc	string	3	Sigla da filial, sem espaços e sem caracteres especiais	Sim
	chave_nfe	int	1		Não
	regiao	string	1		Não

data (JSON)	tipoveic	char	1	Deve ser: <u>A</u> (veículo agregado) <u>P</u> (veículo próprio)	Sim
	dt_prev_ent	date	AAAA-MM-DD		Não
	cnpj_cliente	int	11	CNPJ ou CPF do cliente	Sim
	uf	string	1	Máximo de 2 caracteres, exp: <u>SP</u> , <u>RJ</u> etc	Sim
	obs	string	1		Não

2.7.1 Exemplo de JSON

```
{
  "romaneio": 4321,
  "codmot": "38245566789",
  "nome_mot": "João da Silva",
  "placa_veic": "PTO-3344",
  "tel_mot": "11 97654-8976",
  "cte": 127800,
  "seq_ent": 1,
  "nf": 654,
  "volume": 4,
  "peso": 345,
  "valor_da_mercadoria": 2345.98,
  "cliente": "NOME DA EMPRESA S.A",
  "email_cliente": "email@dominio.com",
  "nome_do_destino": "MEU DESTINATÁRIO",
  "tel_destino": "11 2345-8907",
  "cidade_de_destino": "São Paulo",
  "tipomov": "ENT",
  "datatransf": "2018-02-20",
  "prazoentrega": "2018-02-20",
  "bairro": "JARDIM VILA FORMOSA",
  "endereco": "R. General Amador 345",
  "filial": "SPO",
  "filial_doc": "SPO",
  "chave_nfe": "37786554345678986754377865543456789867543345",
  "regiao": "ZONA LESTE",
  "tipoveic": "P",
  "dt_prev_ent": "2018-02-20",
  "cnpj_cliente": "34543323456",
  "uf": "SP",
  "obs": "texto livre"
}, {
  "romaneio": 4321,
  "codmot": "78245566789",
  "nome_mot": "Marcos Rocha",
  "placa_veic": "CRT-6348",
  "tel_mot": "11 94567-8967",
  "cte": 456,
  "seq_ent": 1,
  "nf": 0,
  "volume": 1,
  "peso": 0,
  "valor_da_mercadoria": 1235.98,
  "cliente": "NOME DA EMPRESA S.A",
  "email_cliente": "email@dominio.com",
  "nome_do_destino": "MEU DESTINATÁRIO",
  "tel_destino": "11 3456-8907",
  "cidade_de_destino": "São Paulo",
  "tipomov": "COL",
  "datatransf": "2018-02-20",
  "prazoentrega": "2018-02-20",
  "bairro": "JARDIM VILA FORMOSA",
  "endereco": "R. General Amador 123",
  "filial": "SPO",
  "filial_doc": "SPO",
  "chave_nfe": "37786554345678986754377865543456789867543345",
  "regiao": "ZONA LESTE",
  "tipoveic": "P",
  "dt_prev_ent": "2018-02-20",
  "cnpj_cliente": "34543323456",
}
```

```
    "uf": "SP",  
    "obs": "texto livre"  
  }  
}
```

2.8 Padrão de retorno

Toda requisição retorna o seguinte padrão de dados

Parâmetro	Tipo	Observações
success	boolean	Retorna <u>true</u> para sucesso e <u>false</u> para erro
data	array	Dados de retorno, campos solicitados ou informações sobre o erro
msgErro	string	*caso o campo success retorne como <u>false</u> esse campo é retornado com os detalhes do erro da requisição

```
[{  
  "success": "true",  
  "data": {  
    [...demais campos do serviço ]  
  }  
}]
```

OU...

```
[{  
  "success": "false",  
  "data": {  
    "msgErro" : "Campo NF no formato inválido"  
  }  
}]
```