

Primeiros passos com a API da Nuvemshop
Controle de versões
Autenticação
Recursos da API
Caixa abandonado
Cobrança
Blog
Regras de negócio
Carinho
Campos personalizados de categoria
Categoria
Confira
SDK de finalização de compra
Cupons
Cliente
Campos personalizados do cliente
Desconto
Ordem de projeto
Modelos de e-mail
Pedido de Cumprimento
Localização

Verão: 2025-03

Webhook

Um webhook é uma ferramenta que permite receber notificações sobre um determinado evento. Ele permite registrar uma URL <https://> que receberá os dados do evento, armazenados em JSON. Os webhooks podem ser registrados para os seguintes eventos:

Categoria	Eventos
Aplicativo	desinstalado/suspenso/retomado
Categoria	criado/atualizado/excluído
Cliente	criado/atualizado/excluído
Ordem	criado/atualizado/pago/embalado/entregue/cancelado/campos_personalizados_atualizados/editados/pendente/
Produto	criado/atualizado/excluído
Variante do produto	campos_personalizados_atualizados
Dominio	atualizado
Campo personalizado do pedido	criado/atualizado/excluído
Campo personalizado de variante do produto	criado/atualizado/excluído
Subscrição	atualizado
Cumprimento	atualizado

Para se inscrever no evento de um produto criado, por exemplo, você deve enviar `product/created` no campo do evento. Basicamente, você faz uma requisição POST no endpoint de Webhooks, fornecendo um corpo com o tipo de webhook que deseja criar, bem como a URL onde você será notificado.

Agora, digamos que você queria ser notificado sempre que um pedido for pago, para não precisar fazer chamadas agendadas para obter essa informação. Nesse caso, você deve criar um Webhook com o evento `order/paid` e a URL onde a notificação será enviada no seu backend.

Os eventos `app/suspended` e `app/resumed` referem-se à suspensão do acesso à API devido à falta de pagamento.

Não é permitido usar um domínio `localhost:12345/nuvemshop` para webhooks.

Para auxiliar no seu desenvolvimento, você pode usar o [RequestCatcher](#). O PostCatcher é um aplicativo web que gera uma URL exclusiva que você pode usar como um endpoint de webhook e exibir quaisquer solicitações POST enviadas a ele para que você possa analisá-las.

Sobre a ordenação e a idempotência das mensagens

Vale ressaltar que as mensagens de webhook são gerenciadas por meio de um sistema distribuído, sendo processadas por múltiplos servidores independentes que trabalham em paralelo para garantir o processamento rápido, confiável e eficiente das milhares de mensagens enviadas por minuto. Devido à natureza do sistema distribuído, não podemos garantir que a ordem em que as mensagens são enfileiradas para processamento seja a mesma ordem em que são processadas.

Além disso, algumas mensagens podem ser enviadas várias vezes com o mesmo conteúdo. Essas mensagens devem ser tratadas como únicas. É responsabilidade do desenvolvedor que integra seu serviço ao nosso webhook garantir a idempotência.

Regras para

- Mensagens enviadas com corpos idênticos devem ser tratadas como únicas.
- Mensagens enviadas com conteúdo idêntico, mas com atributos diferentes, devem ser tratadas como únicas.
- Mensagens enviadas com conteúdo diferente (por exemplo, número de tentativas incluído no corpo da mensagem) devem ser tratadas como duplicadas.

Propriedade	Explicação
eu_id	O identificador numérico exclusivo para o Webhook
URL	A URL para onde o webhook deve enviar a solicitação POST quando o evento ocorrer. Deve ser HTTPS .
evento	O evento que irá acionar o webhook
criado_em	Data de criação do Webhook no formato ISO 8601
atualizado_em	Data da última atualização do Webhook no formato ISO 8601

Verificando um webhook

Os webhooks criados por meio da API podem ser verificados através do cálculo de uma assinatura digital.

Cada solicitação de webhook inclui um `x-linkeditstore-hmac-sha256` cabeçalho gerado usando o segredo do aplicativo, juntamente com os dados enviados na solicitação.

Observe que algumas estruturas irão alterar o cabeçalho para `HTTP_X_LINKEDSTORE_HMAC_SHA256`.

Veja o exemplo em PHP abaixo:

```
<?php
define('APP_SECRET', 'secret');

function verify_webhook($data, $shmac_header) {
    return $shmac_header == hash_hmac('sha256', $data, APP_SECRET);
}

$shmac_header = $_SERVER['HTTP_X_LINKEDSTORE_HMAC_SHA256'];
$data = file_get_contents('php://input');
$verified = verify_webhook($data, $shmac_header);
echo 'Webhook verified: ' . var_export($verified, true);
```

Sobre a ordenação e a idempotência das mensagens

Regras para Desduplicação

Propriedades

Verificando um webhook

Parâmetros

aplicativo/desinstalado

categoria/criada

categoria/atualizada

categoria/excluída

cliente/criado - cliente/atualizado - cliente/excluído

pedido/criado - pedido/atualizado - pedido/pago - pedido/embalado - pedido/concluído -

pedido/cancelado -

pedido/campos_personalizados_atualizados -

pedido/editado -

pedido/pending -

produto/criado - produto/atualizado -

produto/editado

variante_do_produto/campos_personalizados_atualizados

domínio/atualizado

order_custom_field/created -

order_custom_field/updated -

order_custom_field/deleted

product_variant_custom_field/create

d -

product_variant_custom_field/update

product_variant_custom_field/delete

- **Id**: ID do aplicativo.

categoria/criada - categoria/atualizada - categoria/

- **Id**: identificador da categoria.

cliente/criado - cliente/atualizado - cliente/

- **Id**: identificação do cliente.
- **event_launch_ts**: Data e hora em que o evento foi iniciado.

pedido/criado - pedido/atualizado - pedido/pago - pedido/embalado - pedido/concluido - pedido/cancelado - pedido/campos_personalizados_atualizados - pedido/editado - pedido/pendente - pedido/

- **Id**: ID do pedido.

produto/criado - produto/atualizado - produto/

- **Id**: identificação do produto.

variente_do_produto/

- **Id**: ID da variante do produto.

dominio/

Nenhum parâmetro adicional é enviado com este evento. Para obter a lista de domínios, consulte os recursos [da Loja](#).

order_custom_field/created - order_custom_field/updated - order_custom_field/

- **Id**: ID do campo personalizado.

product_variant_custom_field/created - product_variant_custom_field/updated - product_variant_custom_field/

- **Id**: ID do campo personalizado.

assinatura/

- **concept_code**: Conceito.
- **service_id**: ID do serviço.
- **event_launch_ts**: Data e hora em que o evento foi iniciado.

de exemplo do webhook

```
{
  "store_id":123,
  "event": "product/created",
  "id":1948209
}
```

de repetição

Uma notificação Webhook espera um código de status 2XX como resposta (para garantir que o aplicativo recebeu a notificação) e tem um tempo limite de 10 segundos. Ou seja, aguardamos 10 segundos por um código 2XX como resposta (pode ser qualquer código que comece com 2 - 200/201, etc.).

Quando uma notificação webhook não pode ser entregue devido a um problema (por exemplo, quando o aplicativo está inativo), a política de repetição é ativada.

As primeiras 4 tentativas são: uma imediatamente após o tempo limite e, em seguida, aproximadamente aos 5, 10 e 15 minutos. Depois disso, inicia-se um recuo exponencial, multiplicando o tempo de espera da tentativa anterior por 1,4 nas próximas 48 horas (até 18 tentativas).

Visão geral do processo:

- Enviamos o webhook.
- Aguardamos 10 segundos pela confirmação da resposta do aplicativo.
- Caso não recebemos nenhuma resposta, a política de repetição será ativada.

Um código de resposta bem-sucedido interromperá as próximas notificações. Mas, se isso não acontecer, as notificações serão enviadas 18 vezes.

obrigatórios

Para que nossa empresa esteja em conformidade com as leis de proteção de dados (como a [LGPD no Brasil](#)), utilizamos Webhooks que facilitam a comunicação entre consumidores, comerciantes e aplicativos integrados, a fim de acionar notificações de remoção ou atualização de dados compartilhados, quando recebemos uma solicitação.

Basicamente, a LGPD obriga as plataformas tecnológicas a fornecer mecanismos através dos quais os proprietários dos dados possam:

- Solicite um relatório da plataforma para descobrir quais dados pessoais dessa pessoa ela possui.
- Solicite à plataforma a exclusão dos seus dados pessoais.

Os aplicativos da Tiendanube são plataformas digitais que podem armazenar dados tanto de comerciantes quanto de consumidores. Por isso, os parceiros também têm a responsabilidade de cumprir essa lei. Eles são obrigados a excluir os dados ou a informar quais dados possuem, sempre que solicitado.

Para que os parceiros possam ficar sabendo dessas ordens, a Tiendanube criou esse mecanismo de webhooks. Os webhooks são notificações automáticas que a Tiendanube envia aos parceiros quando ocorrem eventos específicos que exigem a exclusão ou o envio de dados. Existem três tipos de webhooks:

URL webhook store redact
URL webhook customers redact
URL webhook customers data request

armazenar/

Solicitação para excluir informações da loja. Após um lojista desinstalar seu aplicativo, o Nuvemshop|Tiendanube envia este webhook com o ID da loja para que você possa excluir as informações do lojista do seu banco de dados. Payload:

```
{
  "store_id": 123;
}
```

clientes/

Solicitação para apagar informações do consumidor. Se o consumidor não fez nenhum pedido na loja nos últimos 6 meses, o webhook será enviado após 5 dias. Se o cliente fez um pedido nos últimos 6 meses, o pedido excluído será mantido até que 6 meses se passem. Payload:

```
{
  "store_id": 123,
  "customer": {
    "id": 1,
    "email": "email@email.com",
    "phone": "+55...",
    "identification": ...
  },
  "orders_to_redact": [213, 3415, 21515]
}
```

clientes/

Solicitação de informações do cliente. É responsabilidade do aplicativo enviar as informações diretamente ao comerciante. Carga útil:

```
{
```

```

store_id: 123,
customer: {
    id: 1,
    email: email@email.com
    phone: +05...
    identification: ...
},
orders_requested: [213, 3415, 21515],
checkouts_requested: [214, 3416, 21518],
drafts_orders_requested: [10, 1245, 5456],
data_request: {
    id: 456,
}
}

```

GET /

Receba uma lista de todos os Webhooks para sua aplicação.

Parâmetro	Explicação
desde_id	Restringir resultados a depois do ID especificado
URL	Exibir tags de webhook com uma URL específica
evento	Exibir webhooks com um evento específico
criado_em_min	Exibir webhooks criados após a data (formato ISO 8601)
criado_no_máximo	Exibir webhooks criados antes da data (formato ISO 8601)
atualizado_ao_min	Exibir webhooks atualizados pela última vez após a data (formato ISO 8601)
atualizado_no_máximo	Exibir a data da última atualização dos webhooks (formato ISO 8601)
página	Página a ser exibida
por página	Quantidade de resultados
campos	Lista de campos a serem incluídos na resposta, separados por vírgulas.

GET /

HTTP/1.1 200 OK

```
[
  {
    "created_at": "2013-01-03T09:11:51-03:00",
    "event": "app/uninstalled",
    "id": 101,
    "updated_at": "2013-03-11T09:14:11-03:00",
    "url": "https://myapp.com/uninstall"
  },
  {
    "created_at": "2013-04-07T09:11:51-03:00",
    "event": "order/created",
    "id": 5676,
    "updated_at": "2013-04-08T11:11:51-03:00",
    "url": "https://myapp.com/order_created_hook"
  }
]
```

GET /webhooks?since_id=

HTTP/1.1 200 OK

```
[
  {
    "created_at": "2013-04-07T09:11:51-03:00",
    "event": "order/created",
    "id": 5676,
    "updated_at": "2013-04-08T11:11:51-03:00",
    "url": "https://myapp.com/order_created_hook"
  }
]
```

GET /webhooks/{id}

Receba um único Webhook

Parâmetro	Explicação
campos	Lista de campos a serem incluídos na resposta, separados por vírgulas.

GET /webhooks/

HTTP/1.1 200 OK

```
{
  "created_at": "2013-04-07T09:11:51-03:00",
  "event": "order/created",
  "id": 5676,
  "updated_at": "2013-04-08T11:11:51-03:00",
  "url": "https://myapp.com/order_created_hook"
}
```

POST /

Criar um novo Webhook

POST /

```
{
  "url": "foobar",
  "event": "invalid_event"
}
```

HTTP/1.1 422 Unprocessable Entity

```
{
  "event": [
    "Invalid event specified. Events allowed are available on Parameters section: https://tiendanube.gi...
  ],
  "url": ["invalid url specified"]
}
```

POST /

```
{
  "event": "product/created",
  "url": "https://myapp.com/product_created_hook"
}
```

HTTP/1.1 201 Created

```
{
  "id": 5677,
  "event": "product/created",
  "url": "https://myapp.com/product_created_hook"
}
```

```
        "created_at": "2013-06-01T15:12:15-03:00",
        "event": "product/created",
        "id": 8901,
        "updated_at": "2013-06-01T15:12:15-03:00",
        "url": "https://myapp.com/product_created_hook"
    }
```

PUT /webhooks/{id}

Modificar um Webhook existente

PUT /webhooks/

```
{
    "created_at": "2013-04-07T09:11:51-03:00",
    "event": "category/created",
    "id": 5678,
    "updated_at": "2013-04-08T11:11:51-03:00",
    "url": "https://myapp.com/category_created_hook"
}
```

HTTP/1.1 200 OK

```
{
    "created_at": "2013-04-07T09:11:51-03:00",
    "event": "category/created",
    "id": 5679,
    "updated_at": "2013-06-01T12:11:14-03:00",
    "url": "https://myapp.com/category_created_hook"
}
```

EXCLUIR /webhooks/{id}

Remover um Webhook

EXCLUIR /webhooks/

HTTP/1.1 200 OK

{}

Última atualização em 29 de janeiro de 2026

Anterior
« Transação

Próximo
Gulás »

Documentos

Designers Web ↗
Nimbus ↗

Comunidade

Instagram ↗
Linkedin ↗
Twitter ↗
YouTube ↗
Facebook ↗

Mais

Parceiros ↗
Termos e Condições ↗
Política de Privacidade ↗
Política de qualidade ↗
Apolar

Copyright © 2026 Nuvemshop. Desenvolvido com Docusaurus.