

Documentação técnica dos webhooks

Webhook é uma forma de integração entre dois sistemas onde um sistema notifica o outro quando um evento acontece.

No caso da Foody Delivery, os webhooks são muito úteis quando um sistema integrado à Foody Delivery quer receber informações sobre atualizações (notificações) sobre os pedidos.

Exemplo: quando um pedido foi despachado para um motoboy ou entregue para o cliente.

Boas práticas no uso dos hooks

As chamadas dos webhook da Foody Delivery tem um **timeout de 10 segundos**, então evite grandes processamentos diretamente na URL que é chamada pelo hook.

A URL invocada pelo webhook deve **sempre retornar http-status 200**, qualquer status diferente de 200 será considerado um erro na chamada do hook.

IMPORTANTE: o webhook não tem lógica ou inteligência para tratar erros como 400, 401, 404 retornados pelo seu endpoint. A missão do hook é entregar o evento para o seu endpoint, então se seu endpoint foi chamado com sucesso e você recebeu o evento, você deve retornar http status como 200 pois você recebeu o evento.

Quando o hook encontra um erro (http-status diferente de 200), ele tenta mais uma vez depois de 60-120 segundos.

AVISO! Se seu endpoint retornar consecutivamente mais de 10 erros, o webhook será automaticamente desativado.

Se você precisa de uma **ferramenta de apoio** para desenvolver sua API (URL invocada pelos hooks) localmente e ainda receber as chamadas normalmente, nós recomendamos que você use a ferramenta [ngrok](#).

Como configurar e ativar os webhooks

Acesse **Minha Conta > Configurações Avançadas > Criar Gatilho**, então entre com os seguintes dados:

- A URL a ser invocada pelo hook.

- Clique em **Ativado** para ativar ou desativar o hook.
- O token de autorização é gerado automaticamente, mas você pode alterá-lo se quiser.

Editar Gatilho

Ativado

URL:

https://meusite.com/hooks/myhook

Token de autorização:

67827ec3da1847feb09c3886aebd8acf

Salvar Cancelar

O payload enviado pelo hook contém o token de autorização configurado previamente para que você autentique a chamada do hook.

Exemplo de payload:

```
{  
  
  "uid": "4228df3a-a54a-40d7-b30f-1385eca4c3dd",  
  
  "updateDate": "2020-06-19T17:42:12Z",  
  
  "event": "order_status_updated",  
  
  "token": "2b100082b90d4ad9b4fezzed960cf652",  
  
  "status": "dispatched"  
}
```

AVISO! Na hora de criar ou atualizar um gatilho, o sistema valida a url do gatilho enviando um payload de teste com o evento 'hook_test'. Desta forma, a URL deve estar disponível e preparada para receber este payload de teste e retornar http-status 200.

Caso a url não retorne http-status 200 no momento desta validação, não será possível proceder com a configuração deste gatilho.