

Distribuição

Webhook

Um webhook é uma requisição `HTTP` enviada automaticamente quando um evento acontece. Isso permite que o próprio servidor notifique o cliente do evento em vez de o cliente perguntar repetidamente. No App de distribuição, webhooks são usados para notificar o cliente quando os mesmos recebem novas distribuições.

Para receber webhooks do App de distribuição é necessário disponibilizar um endpoint para receber os disparos, que são requisições `POST`

Todas requisições internas precisam ter a propriedade `X-Oystr-Auth` **no `headers`, recebendo o token que pode ser adquirido nas configurações do painel do Console como chave de API.**

Ao receber novas distribuições, os webhooks notificarão a url cadastrada com o seguinte corpo.

```
{
  "webhookId": "id",
  "entity": {
    "type": "user",
    "id": "id",
    "username": "username"
  },
  "lastUpdatedAt": null,
  "lastViewedAt": null,
  "endpoints": {
    "result": "/webhook/:id/results",
    "callback": "/webhook/:id/callback"
  }
}
```

CAMPO	DESCRIÇÃO
webhookId	Id do webhook
entity	Dados internos
lastUpdatedAt	Última notificação realizada
lastViewedAt	Última visualização de novas distribuições
endpoints	Endpoints de auxílio para visualização dos resultados

Ajuda

Para ter acesso às rotas da integração, é possível fazer uma requisição `GET` com final `/help`

```
GET: https://console4.oysttr.com.br/api/v1/service/disthook/webhook/help
```

RESPOSTA

```
[
  {
    "method": "GET",
    "endpoint": "/webhook/help",
    "description": "Descrição de cada rota"
  },
  {
    "method": "GET",
    "endpoint": "/webhook/:id",
    "description": "Informações do webhook"
  },
  {
    "method": "POST",
    "endpoint": "/webhook/:id/results",
    "description": "Resultado do webhook e confirmação de visualização"
  },
  {
    "method": "GET",
    "endpoint": "/webhook/:id/callback",
    "description": "Verificação do webhook"
  },
  {
```

```
[
  {
    "method": "GET",
    "endpoint": "/webhook",
    "description": "Busca de webhooks"
  },
  {
    "method": "POST",
    "endpoint": "/webhook",
    "description": "Criação de webhook"
  },
  {
    "method": "PUT",
    "endpoint": "/webhook/:id",
    "description": "Modificação de webhook"
  },
  {
    "method": "DELETE",
    "endpoint": "/webhook/:id",
    "description": "Remoção de webhook"
  }
]
```

Informações

GET: <https://console4.oysttr.com.br/api/v1/service/disthook/webhook/:id>

Para ter acesso às informações de um webhook já cadastrado, é possível fazer uma requisição GET com final `/:id`. O retorno trará as seguintes informações:

```
{
  "id": "id",
  "carteiraId": null,
  "userId": "id",
  "needView": false,
  "needNotify": false,
  "lastStatusCode": null,
  "lastResponse": null,
  "lastUpdateAt": null,
}
```

```
"lastViewedAt": null,
"url": "url",
"headers": "{}",
"createdAt": "2024-01-04T19:06:40.870Z",
"updatedAt": "2024-01-04T19:06:40.870Z",
"deletedAt": null,
"entity": {
  "type": "user",
  "id": "id",
  "username": "username"
}
}
```

CAMPO	DESCRIÇÃO
id	Id do webhook
carteiralId	Id da carteira vinculada
userId	Id do usuário vinculado
needView	Booleano para a necessidade de visualização de novos dados
needNotify	Booleano para a necessidade de notificação no endpoint definido
lastStatusCode	Último statusCode da requisição de notificação feita
lastResponse	Última resposta da requisição de notificação feita
lastUpdateAt	Última notificação realizada
lastViewedAt	Última visualização de novas distribuições
url	Url `POST` de disparo de notificação
headers	Parâmetros para a url de disparo de notificação
createdAt	Data de criação do webhook
updatedAt	Data de atualização do webhook
deletedAt	Data de deleção do webhook
entity	Dados internos

Resultados (novas distribuições)

POST: <https://console4.oyster.com.br/api/v1/service/disthook/webhook/:id/results>

Para ter acesso às novas distribuições que foi notificado por um webhook, é possível fazer uma requisição `POST` com final `/:id/results`. **Apenas serão mostradas distribuições recebidas após a última notificação, distribuições notificadas anteriormente e que não foram vistas não aparecerão.** O retorno trará as seguintes informações:

```
{
  "id": "id",
  "dataDistribuicao": "2024-01-10T11:23:24-03:00",
  "termo": "Lorem",
  "tribunal": "LI",
  "numeroProcesso": "00000000-00.0000.0.00.0000",
  "tipoOcorrencia": "Lorem",
  "reu": "Lorem Ipsum",
  "autor": "Lorem Ipsum",
  "forum": "LI-4",
  "vara": "2ª Lorem Ipsum",
  "cidade": "LOREM",
  "uf": "LI",
  "valorOcorrencia": "20000.000",
  "advogadoautor": "Lorem Ipsum",
  "publicacao": null,
  "linkDocumentosIniciais": "https://loremipsum.com.br/lorem.pdf",
  "processoEletronico": "00000000-00.0000.0.00.0000",
  "tipoProcesso": "Lorem Ipsum",
  "instancia": "1",
  "assunto": "Lorem Ipsum",
  "juiz": null,
  "createdAt": "2024-01-11T13:35:21.420Z"
}
```

Sem o parâmetro `all` passado via query, direto pela url, esse endpoint apenas retornará as distribuições novas. Passando o parâmetro citado, o endpoint passará a retornar todas as distribuições do usuário ou da carteira, dependendo do tipo de vínculo do webhook, dessa forma, é possível filtrar elas com os seguintes parâmetros, esses passados pelo `body`.

```
{
  "from": "0",
  "size": "20",
  "calendarInterval": "day",
  "year": "2024",
  "month": "1",
  "day": "1"
}
```

CAMPO	DESCRIÇÃO	**Valor padrão**
from	Índice inicial de corte para retorno das distribuições (paginador)	0
size	Índice final de corte para retorno das distribuições (paginador)	20
calendarInterval	Tipo de data de filtro das distribuições (year, month, day)	year
year	Ano da distribuição	null
month	Mês da distribuição	null
day	Dia da distribuição	null

Verificação

GET: <https://console4.oyster.com.br/api/v1/service/disthook/webhook/:id/callback>

Para saber se há novas distribuições a serem visualizadas, é possível fazer uma requisição GET com final `/:id/callback`. O retorno trará a seguinte informação:

```
{
  "needView": false
}
```

Busca

GET: <https://console4.oyster.com.br/api/v1/service/disthook/webhook>

Para buscar por webhooks já cadastrados na conta, é possível fazer uma requisição GET com final `/`.

Criação

POST: <https://console4.oysttr.com.br/api/v1/service/disthook/webhook>

Para criar um webhook, é possível fazer uma requisição `POST` com final `/`. O `body` terá o seguinte formato:

```
{
  "url": "url",
  "headers": {},
  "carteiralId": "id"
}
```

CAMPO	DESCRIÇÃO
url	Url <code>POST</code> de disparo de notificação (obrigatório)
headers	Parâmetros para a url de disparo de notificação (opcional)
carteiralId	ID interno da carteira que deseja notificação

Atualização

PUT: <https://console4.oysttr.com.br/api/v1/service/disthook/webhook/:id>

Para atualizar um webhook, é possível fazer uma requisição PUT com final `/:id`. O body terá o seguinte formato:

```
{
  "url": "url",
  "headers": {}
}
```

Deleção

DELETE: <https://console4.oysttr.com.br/api/v1/service/disthook/webhook/:id>

Para deletar um webhook, é possível fazer uma requisição `DELETE` com final `/:id`

Revision #2

Created 25 March 2024 02:50:24 by Luan Melo

Updated 25 March 2024 03:26:03 by Luan Melo