

# API 1:1

## CREDENCIAL

POST: <https://api4.oyster.com.br/credentials/validate/:botId/v3.0.0-dev>

## USUÁRIO E SENHA (UP)

objeto **“credentials”**, com par **usuário e senha**, será utilizado pelo robô para conseguir acesso ao sistema no qual ele irá executar as tarefas. Por exemplo, em um robô de tribunal onde a tarefa é obter a última movimentação processual, esse usuário/senha será utilizado para entrar no sistema do tribunal.

Quando informado na requisição (via arquivo **payload** ou especificado no **body** da requisição), é necessário seguir o formato abaixo:

```
{
  "username": "...",
  "password": "...",
  "credentialsOption": "..."
}
```

## Parâmetros

Parâmetro	Tipo	Descrição
<b>username</b>	<i>String (obrigatório)</i>	Usuário utilizado para acessar o sistema em que o robô irá executar a tarefa.
<b>password</b>	<i>String (obrigatório)</i>	Senha correspondente ao usuário informado.
<b>credentialsOption</b>	<i>String (opcional)</i>	Valor opcional que será informado indicando particularidades das opções informadas. Quando necessário, a documentação do robô irá apontar.

## CERTIFICADO A1 (CP)

O objeto **“credentials”** com certificado A1 será utilizado pelo robô para conseguir acesso ao sistema no qual ele irá executar as tarefas. Por exemplo, em um robô unificado intimações onde a tarefa é obter a todas as intimações disponíveis para o advogado em diversos tribunais do Brasil, esse certificado A1 será utilizado para entrar no sistema de cada tribunal especificado.

Quando informado na requisição (via arquivo ***payload*** ou especificado no ***body*** da requisição), é necessário seguir o formato abaixo:

```
{
  "username": "...",
  "base64Cert": "...",
  "pin": "...",
  "credentialsOption": "..."
}
```

## Parâmetros

Parâmetro	Tipo	Descrição
<b>username</b>	<i>String (obrigatório)</i>	Usuário utilizado para acessar o sistema em que o robô irá executar a tarefa.
<b>base64Cert</b>	<i>String (obrigatório)</i>	Conteúdo do certificado A1 codificado em <a href="#">Base64</a>
<b>pin</b>	<i>String (obrigatório)</i>	PIN ou senha utilizada para utilizar o certificado.
<b>pin</b>	<i>String (opcional)</i>	Valor opcional que será informado indicando particularidades das opções informadas. Quando necessário, a documentação do robô irá apontar.

# Criação da Execução 1:1

POST: <https://api4.oysttr.com.br/v1/execute/single>

## Código de retorno

Retorno	Descrição
202	inicia a execução de um item. Retona o id para consulta futur
5xx ou 4xx	Falha ao iniciar a execução do item. Nestes casos é necessário utilizar (manualmente via suporte) o header X-Oysttr-TraceId retornado para verificar se o item foi recebido pela API e está em execução ou não.

## Formato da Requisição (via HTTP)

```

{
  "dry": false,
  "bot": "",
  "version": "",
  "cid": "",
  "timeout": "",
  "deadline": "",
  "credentials": {
    "username": "",
    "password": ""
  },
  "data": {},
  "files": [{
    "bound": true,
    "property": "",
    "description": "",
    "name": "",
    "data": ""
  }]
}

```

CAMPO	Obrigatório	Formato	Default	Descrição
dry	Não	Boolean	false	Indica se é uma execução de teste
bot	Sim	String		Id do robô
version	Sim	String		Versão do robô
cid	Não	String	vazio	Identificador único da requisição <sup>1</sup>
force	Não	Boolean	vazio	Indica se um item único, discriminado pelo cid deve ser reexecutado ou não
timeout	Não	Duration	5 minutos	Tempo máximo aguardando o retorno de um item desde o início da execução <sup>2</sup> . Exemplo: 30s
deadline	Não	ZonedDateTime	vazio	Data máxima para o início da execução do item <sup>3</sup> no formato ISO 8601. Exemplo 2022-01-01T00:00:00.0-03:00
credentials	Robô	json		Credenciais usadas na requisição
data	Sim	json		Dados passados diretamente para o robô. Cada robô possui um formato específico.
file	Robô	json		Arquivos passados para o robô

1. ID usado para evitar requisições duplicadas. Caso o cid não seja único a resposta será do tipo Duplicate . Caso o cid seja inválido a requisição será rejeitada
2. . Após este período a resposta será do tipo Timeout
3. Caso a execução do ítem não tenha iniciado até esta data a resposta será do tipo Overdue.

## Retorno da chamada (assíncrono)

GET: <https://api.oysttr.com.br/v1/execute/single/:bot/:cached?format=latest>

### Código de retorno

Retorno	Descrição
404	id inválido ou ítem rejeitado. Nunca haverá uma resposta para este item.
202	Reposta indisponível, aguarde
200	Resposta disponível

### Resultados

Os resultados ficam disponíveis no endereço `/execute/single/[bot-id]/[id]?format=[legacy|latest]` de acordo com o header `X-Oysttr-ReturnPath`.

O payload recebido depende do valor da variável forma

**Formato Legado** QueryString: `format=legacy`

```
{
  "type": "",
  "payload": {}
}
```

**Formato Recomendado** QueryString: `format=latest`

O payload em json obtido quando a resposta está disponível segue o seguinte modelo:

```
{
  "account": 0,
  "user": 0,
  "bot": "",
```

```
"version": "",
"cid": "",
"request": "",
"started": "",
"ended": "",
"taskTime": "",
"timeout": "",
"finishedAs": "",
"retry": "",
"dry": false,
"result": {}
}
```

Campo	Formato	Descrição
account	Long	Conta que realizou a requisição
user	Long	Usuário que realizou a requisição
bot	String	Robô
version	String	Versão do robô
cid	String	id de integração
request	String	Id da requisição
started	ZonedDateTime	Data de início da execução do item no formato ISO-8601
ended	ZonedDateTime	Data de início da execução do item no formato ISO-8601
taskTime	Duration	Tempo de execução do item
timeout	Duration	Timeout efetivamente usado
retry	Enum	indica se é possível re-executar o item
dry	Json	Indica se é uma execução DE TESTE
finishedAs	Enum	Tipo de resultado (tabela abaixo)
result	Json	Payload com o resultado

# Tipos de retorno

Os tipos de resultados possíveis:

Resposta	Permite Reexecução	Origem	Descrição
Response		Bot	Resposta válida
PartialResponse			
NotAllowed		Bot	Não é permitida a execução do item
NotFound		Bot	Item não encontrado
NotConsistent		Bot	Dados inconsistentes
NotAvailable		Bot	Item não disponível
ProxyError	Sim	Bot	Erro de proxy
CaptchaError	Sim	Bot	Erro de captcha
Forbidden		Bot	Execução do item não é permitida
BotError		Bot	Erro no robô
Other	Não	Bot	Outros
RequestHasViolations	Não	Infra	Pedido contém informações inválidas. O item não foi executado.
NotAndApiRequest	Não	Infra	Formato inválido de requisição. O item não foi executado
NotAccepted	Não	Infra	Execução do item não foi aceita. O item pode ou não ter sido executado
Timeout	Não	Infra	Timeout durante a execução do item. O item pode ou não ter sido executado
Overdue	Não	Infra	Execução do item ocorreria após o deadline. Item não foi executado
Duplicate	Não	Infra	Item duplicado baseado no cid. Item não foi executado
UnexpectedError	Não	Infra	Erro inesperado. O item pode ou não ter sido executado
Unknown	Não	Infra	Desconhecido. O item pode ou não ter sido executado

## Requisições Duplicadas

A Oystr não testa os dados enviados para a execução com a exceção dos atributos cid e force . Caso o cid esteja presente na requisição, a Oystr fará a verificação se este cid foi executado nos últimos 15 dias para esta conta e robô. Caso o cid seja "duplicado", o ítem será executado apenas se o parâmetro force estiver presente com o valor true , caso contrário a resposta do ítem será do tipo Duplicate .

A Oystr emprega uma estratégia estatística para verifica ser o item já foi executado. Esta estratégia permite falsos positivos, ou seja:

1. item não foi executado (com certeza) para esta conta e robô nos últimos 15 dias, ou
2. . item possivelmente já foi executado antes (false positivo)

Um cid , se presente, deve ter entre 1 e 50 caracteres alfanuméricos ou '-' de acordo com a seguinte expressão regular: `[0-9a-zA-Z\\-]{1,50}` . Caso o cid seja inválido o item não será executado, a requisição será ignorada e ficará sem resposta, retornando 404 em consultas futuras.

## Reexecução

Em alguns casos é possível que um item seja reexecutado em caso de erro, mesmo que um cid tenha sido utilizado. Os critérios que definem se um item pode ou não ser reexecutado ficam a cargo do cliente da API que deve observar os seguintes critérios:

1. Se houve uma execução anterior "do mesmo" ítem, esta deve ter retornado como insucesso, com o parâmetro `retry = SAFE` . Caso contrário não é seguro reexecutar o ítem.
2. . Caso um cid seja utilizado, ele deve ser único ou o parâmetro `force = true` deve ser utilizado para ignorar o teste de duplicidade.

A Oystr não verifica se a requisição anterior retornou com `retry = SAFE` durante a nova execução, mesmo que o cid seja o mesmo. Esta verificação fica a cargo do cliente/usuário da api.

---

Revision #2

Created 25 March 2024 03:17:51 by Luan Melo

Updated 3 April 2024 15:12:28 by Luan Melo