Dotpeenator®

BY SERMATEK

GUIA DE INTEGRAÇÃO

Documentação da API RESTful.

Endpoints HTTP para as máquinas de marcação por micropercussão Dotpeenator®, permitindo que sistemas MES e ERP enviem dados de marcação, gerenciem arquivos de modelo e controlem o processo de marcação.

PROTOCOLO	FORMATO DE DADOS	MÉTODOS	PORTA PADRÃO
HTTP/1.1	JSON (UTF-8)	GET / POST	8080

0 Visão Geral

A API RESTful Dotpeenator® é executada na unidade de controle da máquina e fornece endpoints HTTP para o sistema de marcação, permitindo que sistemas MES realizem tarefas como enviar dados de marcação, gerenciar arquivos e controlar a máquina de marcação.

- **Protocolo:** HTTP/1.1
- **Formato de Dados:** JSON (codificação UTF-8)
- **Métodos de Requisição:** GET / POST
- **Endereço Padrão:** `http://127.0.0.1:8080` na própria unidade de controle. A partir de outro computador da rede, use o endereço IP da máquina, ex.: `http://<IP-da-máquina>:8080`.

Nos caminhos dos endpoints, nomes de campos e mensagens (por exemplo `/api/printer/start`, `printerStatus`, `printer busy`), a palavra *printer* refere-se à máquina de marcação Dotpeenator®. Esses identificadores devem ser usados exatamente como estão escritos.

— Informações Gerais

CÓDIGOS DE STATUS HTTP

CÓDIGO	DESCRIÇÃO
`200`	Requisição bem-sucedida; veja o JSON de resposta
`400`	JSON inválido no corpo da requisição
`404`	Endpoint não encontrado
`405`	Método não permitido
`411`	A requisição POST não possui o cabeçalho `Content-Length`

FORMATO DE RESPOSTA

Todas as respostas da API estão em formato JSON com a seguinte estrutura geral:

```
{
    "status": "ok",
    "message": "successfully"
}
```

- `status` : `"ok"` ou `"error"`
- `message` : Descrição do resultado
- Alguns endpoints incluem campos adicionais (ex.: `data` , `file` , `printerStatus`)

EXEMPLO DE REQUISIÇÃO (CURL)

```
curl -X POST http://127.0.0.1:8080/api/setData \
  -H "Content-Type: application/json" \
  -d '{"data": [...]}'
```

Endpoints de Dados de Marcação

1.1 Enviar Dados de Marcação `POST` `/api/setData`

Envia conteúdo de texto para os objetos de marcação especificados, habilita/desabilita a marcação e controla a visibilidade.

> **Pré-requisito:** A máquina deve estar no estado `idle` (ociosa).

PARÂMETROS DA REQUISIÇÃO

CAMPO	TIPO	OBRIGATÓRIO	PADRÃO	DESCRIÇÃO
`data`	Array	**Sim**	—	Array de objetos de dados de marcação

CAMPOS DE CADA OBJETO EM DATA

CAMPO	TIPO	OBRIGATÓRIO	PADRÃO	DESCRIÇÃO
`name`	String	**Sim**	—	Nome do objeto de marcação (deve coincidir com o nome no modelo)

CAMPO	TIPO	OBRIGATÓRIO	PADRÃO	DESCRIÇÃO
`text`	String	Não	—	Conteúdo de texto do objeto. Se omitido, apenas `enable` / `visible` são aplicados
`part`	Int	Não	`0`	Índice do codificador (para objetos com vários segmentos de conteúdo)
`enable`	Bool	Não	`true`	Define se este objeto participa da marcação
`visible`	Bool	Não	`true`	Define se este objeto está visível

EXEMPLO

```
curl -X POST http://127.0.0.1:8080/api/setData \
  -H "Content-Type: application/json" \
  -d '{
     "data": [
         {
             "name": "SN",
             "text": "202405120001",
             "part": 0
         },
         {
             "name": "DateCode",
             "text": "2024-05-12",
             "enable": true,
             "visible": true
         },
         {
             "name": "Logo",
             "enable": false
         }
     ]
  }'
```

RESPOSTA DE SUCESSO (TODOS OS OBJETOS PROCESSADOS COM SUCESSO)

```
{
    "status": "ok",
    "message": "successfully",
    "data": [
        {
            "name": "SN",
            "text": "202405120001",
            "status": "ok",
            "message": "successfully"
        },
        {
            "name": "DateCode",
            "text": "2024-05-12",
            "status": "ok",
            "message": "successfully"
        },
        {
            "name": "Logo",
            "text": "",
            "status": "ok",
            "message": "successfully"
        }
    ]
}
```