



Tribunal de Justiça do Estado da Paraíba
GESIS - Gerência de Sistemas

Guia de integração para cartórios

Selo Digital – API REST

V1.0

Sumário

1. Introdução	3
2. Solicitação de registro	3
3. Ambientes	3
4. Autenticação	4
4.1. Obtendo o token	4
5. Serviços auxiliares	5
5.1. Listar tipo de ato	5
5.2. Listar sub-tipo de ato	5
5.3. Listar Natureza Jurídica	6
5.4. Listar tipo de documento registrado	6
6. Serviço enviar atos	7

Histórico de Revisões

	Versão	Descrição	Responsável
20/08/2026	1.0	• Criação do documento.	Kaio Almeida
05/10/2026	1.1	• Adição do tópico “Solicitação de registro”	Júlio Paiva

1. Introdução

Este documento tem como objetivo orientar os cartórios na integração com a **API Selo-Core** do Tribunal de Justiça da Paraíba (TJPB).

A API disponibiliza os serviços necessários para que os cartórios enviem ao TJPB os **atos** por eles emitidos, além de serviços auxiliares que fornecem os dados de domínio (tabelas de apoio) utilizados no preenchimento dos atos.

O principal serviço é o **Enviar Atos**, responsável por receber e processar os Atos emitidos pelo Cartório. Os demais serviços — Tipos de Ato, Sub-Tipos de Ato, Naturezas Jurídicas e Tipos de Documento Registrado — são serviços auxiliares de consulta, utilizados para obter os valores válidos (domínios) que devem ser informados no envio dos Atos.

Todos os serviços exigem autenticação via **token JWT**, emitido pelo Keycloak do TJPB, e estão documentados na página do **Swagger**, que deve ser utilizada como referência complementar a este guia — principalmente para consulta detalhada dos campos, tipos de dados e regras de validação de cada requisição.

Importante: este documento apresenta uma visão geral de cada serviço e exemplos de uso via cURL. Para o detalhamento completo dos contratos (schemas de requisição e resposta), consulte sempre a documentação Swagger do ambiente correspondente.

2. Solicitação de registro ao ambiente de Homologação

É necessário que se abra um chamado de T.I no link: <https://portaldousuariotjpb.lanlink.com.br/> no corpo do chamado deve-se informar o CNPJ, razão social e nome fantasia da empresa, como também o nome do responsável, telefone e email.

3. Ambientes

HOMOLOGAÇÃO

URL Base da API	https://selodigital-cartorios.tjpb.jus.br/selo-core-rest/
URL do Swagger	https://selodigital-cartorios.tjpb.jus.br/selo-core-rest/swagger-ui/index.html
URL do Keycloak	https://keycloak-hml.tjpb.jus.br/login/realms/TJPB/protocol/openid-connect/token

PRODUÇÃO

URL Base da API	https://selodigital.tjpb.jus.br/selo-core-rest/
URL do Swagger	https://selodigital.tjpb.jus.br/selo-core-rest/swagger-ui/index.html
URL do Keycloak	https://keycloak.tjpb.jus.br/login/realms/TJPB/protocol/openid-connect/token

Recomenda-se que a integração seja primeiramente validada no ambiente de **homologação** e, somente após os testes concluídos com sucesso, migrada para o ambiente de **produção**.

Todos os endpoints descritos neste documento devem ser precedidos pela URL base do ambiente utilizado. Por exemplo, o serviço de envio de atos estará disponível em:

<< URL BASE DO AMBIENTE >>/v1/enviar-atos

4. Autenticação

A API utiliza autenticação via **OAuth2 / JWT**, emitido pelo **Keycloak** do TJPB. Cada cartório receberá previamente do TJPB as seguintes credenciais:

- **Client ID** (identificador da aplicação/cartório no Keycloak);
- **Usuário e senha** de acesso.

Essas credenciais devem ser utilizadas para obter um **token de acesso (access token)** junto ao Keycloak. O token retornado deve ser enviado no cabeçalho Authorization de todas as requisições feitas à API, no formato Bearer <token>.

4.1. Obtendo o token

URL e parâmetros de geração de token serão fornecidos pelo TJPB juntamente com as credenciais de acesso (<< PREENCHER URL DO KEYCLOAK/REALM >>).

Exemplo de requisição para geração do token via curl:

```
curl --location '<< URL DO KEYCLOAK
>>/realms/TJPB/protocol/openid-connect/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'grant_type=password' \
--data-urlencode 'client_id=<< CLIENT_ID_DO_CARTORIO >>' \
--data-urlencode 'username=<< USUARIO >>' \
--data-urlencode 'password=<< SENHA >>'
```

Resposta esperada (exemplo simplificado):

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...",
  "expires_in": 300,
  "refresh_expires_in": 1800,
  "refresh_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "Bearer"
}
```

O valor de access_token deve ser utilizado no cabeçalho Authorization das chamadas aos serviços da API, conforme exemplo abaixo:

```
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...
```

Atenção: o token possui tempo de expiração (expires_in). Quando expirado, é necessário solicitar um novo token repetindo o processo acima, ou utilizar o refresh_token, caso disponível.

5. Serviços auxiliares

Os serviços auxiliares abaixo fornecem os domínios (listas de valores) utilizados para preencher corretamente os campos do serviço de Enviar Atos. Todas são requisições GET, exigem token JWT válido e retornam listas em formato JSON.

5.1. Listar tipo de ato

Retorna a lista de todos os tipos de ato disponíveis no sistema.

- **Método/Endpoint:** GET /v1/tipos-ato

```
curl --location '<< URL BASE DO AMBIENTE >>/v1/tipos-ato' \
--header 'Authorization: Bearer << TOKEN >>'
```

Resposta esperada (exemplo):

```
[
  { "id": 10, "descricao": "Registro Civil" },
  { "id": 11, "descricao": "Notas" }
]
```

Para detalhes completos do contrato (campos, tipos e demais respostas possíveis), consulte a operação “**Lista todos os Tipos de Ato**” na página do Swagger.

5.2. Listar sub-tipo de ato

Retorna a lista de sub-tipos de ato vinculados a um determinado tipo de ato, informado por meio do parâmetro tipoAtoId.

- **Método/Endpoint:** GET /v1/sub-tipos-ato?tipoAtoId={id}

```
curl --location '<< URL BASE DO AMBIENTE >>/v1/sub-tipos-ato?tipoAtoId=10' \
--header 'Authorization: Bearer << TOKEN >>'
```

Resposta esperada (exemplo):

```
[
  { "id": 101, "descricao": "Nascimento", "payload":
    "NASCIMENTO_NATIMORTO_OBITO" },
  { "id": 102, "descricao": "Óbito", "payload": "NASCIMENTO_NATIMORTO_OBITO"
  }
]
```

O campo payload indica qual é o tipo de payload (discriminador) que deve ser utilizado no campo payload do Ato ao ser enviado pelo serviço de Enviar Atos (ver seção 5), permitindo ao Cartório identificar antecipadamente qual estrutura de dados deverá ser preenchida para aquele sub-tipo de ato.

O parâmetro tipoAtold é obrigatório e deve corresponder a um id retornado pelo serviço de Listar Tipo Ato (item 4.1). Para detalhes completos, consulte a operação “**Lista Sub-Tipos de Ato filtrados por Tipo de Ato**” na página do Swagger.

5.3. Listar Natureza Jurídica

Retorna a lista de todas as Naturezas Jurídicas ativas no sistema.

- **Método/Endpoint:** GET /v1/naturezas-juridicas

```
curl --location '<< URL BASE DO AMBIENTE >>/v1/naturezas-juridicas' \
  --header 'Authorization: Bearer << TOKEN >>'
```

Resposta esperada (exemplo):

```
[
  { "id": 1, "descricao": "Pessoa Física" },
  { "id": 2, "descricao": "Pessoa Jurídica de Direito Privado" }
]
```

Para detalhes completos, consulte a operação “**Lista todas as Naturezas Jurídicas**” na página do Swagger.

5.4. Listar tipo de documento registrado

Retorna a lista de todos os tipos de documentos registrados ativos no sistema.

- **Método/Endpoint:** GET /v1/tipos-doc-registrado

```
curl --location '<< URL BASE DO AMBIENTE >>/v1/tipos-doc-registrado' \
  --header 'Authorization: Bearer << TOKEN >>'
```

Resposta esperada (exemplo):

```
[
  { "id": 1, "nome": "RG", "descricao": "Registro Geral" },
  { "id": 2, "nome": "CPF", "descricao": "Cadastro de Pessoa Física" }
]
```

Para detalhes completos, consulte a operação “**Lista todos os Tipos de Documento Registrado**” na página do Swagger.

6. Serviço enviar atos

Este é o principal serviço da API. É utilizado pelos cartórios para transmitir ao TJPB os atos por eles emitidos.

- **Método/Endpoint:** POST /v1/enviar-atos
- **Content-Type:** application/json

A requisição é composta por um envelope contendo dados de controle da transmissão (messageID, messageDate, codigoServentia) e uma lista de atos (atos). Cada item da lista representa um Ato e seu campo payload funciona como **discriminador**, indicando qual é o conjunto específico de dados daquele Ato (ex.: NASCIMENTO_NATIMORTO_OBITO, CASAMENTO_HABILITACAO, REGISTRO_IMOVEIS, NOTAS_GENERICO, entre outros). Os valores de domínio utilizados dentro de cada Ato (tipo de ato, sub-tipo de ato, natureza jurídica, tipo de documento) devem ser obtidos previamente através dos serviços auxiliares descritos na seção 4.

Importante: por se tratar de uma estrutura extensa e variável conforme o tipo de Ato, este documento não reproduz o corpo (body) completo da requisição. Para o detalhamento de todos os campos obrigatórios/opcionais de cada tipo de payload, consulte a operação “**Recebe uma lista de atos para processamento**” na página do Swagger.

Exemplo simplificado de chamada via curl (estrutura reduzida, apenas ilustrativa):

```
curl --location '<< URL BASE DO AMBIENTE >>/v1/enviar-atos' \
--header 'Authorization: Bearer << TOKEN >>' \
--header 'Content-Type: application/json' \
--data '{
  "messageID": 1,
  "messageDate": "2026-08-20T10:00:00",
  "codigoServentia": 123456,
  "atos": [
    {
      "payload": "NASCIMENTO_NATIMORTO_OBITO"
      // ... demais campos do Ato conforme o tipo informado – ver
    }
  ]
}'
```

Resposta esperada em caso de sucesso (200):

```
{
  "dataRecebimento": "2026-08-20T10:00:05-03:00",
  "idServentia": 123456,
  "qtdAtosProcessados": 1,
  "dataProcessamento": "2026-08-20T10:00:06-03:00",
  "tamanhoJson": 512,
  "hashJson": "abcdef1234567890"
}
```

Principais respostas de erro:

Código	Situação
400	Payload inválido: campo obrigatório ausente, formato incorreto ou tipo de Ato desconhecido.
401	Token JWT ausente ou inválido.
409	Arquivo já processado anteriormente (retransmissão bloqueada).
500	Erro interno ao processar a requisição.

Para a lista completa dos tipos de payload suportados, os campos obrigatórios de cada um e os exemplos de schema, consulte a página do **Swagger**, na operação de envio de Atos.