

Modelo de Transferência de Dados - Datajud

Modelo de Transferência de Dados

Aplicativo Validador

Modelo de Transferência de Dados - Datajud

Histórico de versão

Versão	Autor	Data	Ação
1.0.0.0	Rosfran Borges	09/03/2020	Criação do Manual
1.0.0.1	Rosfran Borges	18/02/2021	Novas regras de validação
1.0.0.2	Rosfran Borges	19/02/2021	Detalhamento dos códigos de validação
1.0.0.3	Rosfran Borges	10/03/2021	Novos endpoints. Orientações simples para manutenção em containers Docker

Modelo de Transferência de Dados - Datajud

Sumário

1. Introdução	1
2. Configuração	1
3. Uso do Validador	6
3.1.Fazendo uma Requisição – Simulando um Envio de XML.....	6
3.2.Operação de Validação	9
1. ERRO_MOVIMENTO_LOCAL_INVALIDO	12
2. ERRO_MOVIMENTO_NACIONAL_INVALIDO.....	12
3. ERRO_ULTIMO_MOVIMENTO MAIS_3_ANOS.....	12
4. ERRO_ASSUNTO_LOCAL_INVALIDO	12
5. ERRO_ASSUNTO_NACIONAL_INVALIDO.....	12
6. ERRO_ASSUNTO_NAO_FOLHA_INVALIDO.....	13
7. ERRO_SEM_ASSUNTOS.....	13
8. ERRO_MOVIMENTO_COD_PAI_NACIONAL_INVALIDO	13
9. ERRO_MOVIMENTO_NACIONAL_AUDIENCIA_- SEM_COMPLEMENTO_NACIONAL.....	13
10.ERRO_MOVIMENTO_LOCAL_AUDIENCIA_SEM_- COMPLEMENTO_NACIONAL	13
11.ERRO_MOVIMENTO_NACIONAL_REMESSA_SEM_- COMPLEMENTO_NACIONAL	13
12.ERRO_MOVIMENTO_LOCAL_REMESSA_SEM_- COMPLEMENTO_NACIONAL	13
13.ERRO_MOVIMENTO_NACIONAL_NAO_FOLHA_IN- VALIDO	14
14.ERRO_MOVIMENTO_LOCAL_NAO_FOLHA_INVALI- DO	14
15.ERRO_ULTIMO_MOVIMENTO_DATA_INVALIDA.....	14

Modelo de Transferência de Dados - Datajud	
16.ERRO_SEM_MOVIMENTO_CASO_NOVO	14
17.ERRO_CLASSE_PROCESSUAL_INVALIDA	14
18.ERRO_ORGAO_JULGADOR_INVALIDO	14
19.ERRO_SEM_NOME_PARTE	14
20.ERRO_PESSOA_SEM_DOCUMENTO_PRINCIPAL..	15
21.ERRO_PESSOA_SEM_DOCUMENTO.....	15
22.ERRO_PESSOA_COM_DOCUMENTO_CMF_SEM_- NUMERO_DOCUMENTO_PRINCIPAL	15
23.ERRO_PROCESSO_SEM_POLO_ATIVO	15
24.ERRO_GRAU_INVALIDO	15
25.ERRO_NUMERO_PROCESSO_JTR_INVALIDO	15
26.ERRO_NUMERO_PROCESSO	15
27.ERRO_ORGAO_JULGADOR_INVALIDO_NAO_COR- RESPONDE_SIGLA_TRIBUNAL	16
3.3.Outros Endpoints	16
3.3.1./processos/json	16
3.3.2./processos/json/{grauJurisdicao}.....	16
3.3.3./processos/{grauJurisdicao}	16
3.3.4./versão.....	17
3.3.5./validaXSD.....	18
3.4.Instância do Validador Hospedado no CNJ	18
4. Acesso ao código-fonte do projeto Validador	18



Modelo de Transferência de Dados - Datajud

1. Introdução

O objetivo deste aplicativo é permitir que os tribunais executem localmente as suas rotinas de validação de dados XML que serão enviados para o serviço REST do Modelo de Transferência de Dados do CNJ (<https://www.cnj.jus.br/modelo-de-transferencia-de-dados/v1>).

2. Configuração

Para a aplicação funcionar, é necessário instalar o Docker (<https://www.docker.com>). Existem versões do Docker para Windows, Linux e MacOS.

Após instalar o Docker, faça download dos arquivos de configuração dos containers:

URL: https://cnjjustbr-my.sharepoint.com/:u:/g/personal/rosfran_borges_cnj_jus_br/ES2-CHpT5EOdChY_2GlvjwwUBO3hoy22IPoQ84KownCFUKA?e=F2nJu6

Senha: validador2021

Ao descompactar o arquivo, será exibida a estrutura abaixo:

nome	Data de modificação
> build	25 de mar. de 2020 18:18
Changelog.md	10 de fev. de 2021 09:15
docker-compose.yaml	8 de dez. de 2020 18:23
Dockerfile	10 de fev. de 2021 09:15
> sql_create	2 de fev. de 2021 20:24

Nesse diretório há um diretório *build*, no qual está localizada a aplicação, que é um microsserviço feito em Spring Boot e desenvolvido em Java. O arquivo *Changelog.md* tem o histórico de versões do Validador. No diretório *sql_create* estão os

Modelo de Transferência de Dados - Datajud

2

scripts de criação do banco que será usado para as ações de validação dos arquivos XML e enriquecimento (veja Seção 3.2). O arquivo `docker-compose.yaml` é um arquivo de configuração para orquestração de serviços e será necessário para subir a instância MySQL e o container com o backend REST. O arquivo `Dockerfile` é um arquivo que faz o build da imagem da aplicação backend REST.

No mesmo diretório onde foi descompactado e tendo já instalado o Docker (ou o Docker Desktop para usuários MacOS e Windows), pode-se executar o comando seguinte:

docker-compose up

Esse comando subirá 2 containers Docker: uma instância do MySQL (alimentado com os dados das tabelas processuais unificadas e outras tabelas de apoio) e um *backend* com uma interface REST com alguns endpoints implementados. A intenção é que esse *backend* responda da mesma forma que a solução em produção que funciona em <https://www.cnj.jus.br/modelo-de-transferencia-de-dados/>.

Após execução do comando *docker-compose* acima, aparecerão os logs abaixo, que indicam que o processo de subida do Backend REST foi bem sucedido:

```
(base) MacBook-Pro-de-Rosfran:validador rosfranlinsborges$ docker-compose up --build
Creating network "validador_banco" with the default driver
Building app
Step 1/11 : FROM openjdk:11-slim
--> e61f5e080972
Step 2/11 : EXPOSE 8081
--> Using cache
--> 13d9a363de86
Step 3/11 : ENV APP_HOME /app
--> Using cache
--> 7a9c9bca2d01
Step 4/11 : RUN mkdir $APP_HOME
--> Using cache
--> c7a67bb8ac93
Step 5/11 : RUN mkdir $APP_HOME/config
--> Using cache
--> 93cecdad9cf6
Step 6/11 : RUN mkdir $APP_HOME/log
--> Using cache
--> b2b485177824
Step 7/11 : VOLUME $APP_HOME/log
--> Using cache
--> 8cd584a2319f
Step 8/11 : VOLUME $APP_HOME/config
--> Using cache
--> cb99a66731aa
Step 9/11 : WORKDIR $APP_HOME
--> Using cache
--> 7b618b8c399a
Step 10/11 : COPY ./build/libs/*.jar selo-valida-0.0.19.jar
--> e2bb4945dc04
Step 11/11 : ENTRYPOINT ["java", "-XX:+UnlockExperimentalVMOptions", "-Djava.security.egd=file:/dev/./urandom", "-jar", "selo-valida-0.0.19.jar"]
--> Running in 9d9b96080ce4
Removing intermediate container 9d9b96080ce4
--> 96d387b55add
Successfully built 96d387b55add
Successfully tagged validador_app:latest

MacBook-Pro-de-Rosfran:validador rosfranlinsborges$ ping 10.203.1.165
ping: cannot resolve 10.203.1.165: Unknown host
(base) MacBook-Pro-de-Rosfran:validador rosfranlinsborges$ ping 10.203.1.165
PING 10.203.1.165 (10.203.1.165): 56 data bytes
36 bytes from 172.16.0.250: Time to live exceeded
Vr Hl TOS Len ID Flg off TTL Pro cks Src Dst
4 0 00 5400 ddb4 0 0000 00 01 ac1b 10.100.10.4 10.203.1.165
Request timeout for icmp_seq 0
36 bytes from 172.16.0.250: Time to live exceeded
Vr Hl TOS Len ID Flg off TTL Pro cks Src Dst
4 0 00 5400 d874 0 0000 00 01 c8d3 10.100.10.4 10.203.1.165
Request timeout for icmp_seq 1
36 bytes from 172.16.0.250: Time to live exceeded
Vr Hl TOS Len ID Flg off TTL Pro cks Src Dst
4 0 00 5400 7ad5 0 0000 00 01 1e7e 10.100.10.4 10.203.1.165
--- 10.203.1.165 ping statistics ---
5 packets transmitted, 0 packets received, 100.0% packet loss
(base) MacBook-Pro-de-Rosfran:validador rosfranlinsborges$ mysql -h 10.203.2.165
-bash: mysql: command not found
(base) MacBook-Pro-de-Rosfran:validador rosfranlinsborges$
(base) MacBook-Pro-de-Rosfran:validador rosfranlinsborges$
Last login: Wed Feb 3 07:31:42 on console
Restored session: Qua 3 Feb 2021 07:25:09 -03
The default interactive shell is now zsh.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) MacBook-Pro-de-Rosfran:validador rosfranlinsborges$
Last login: Fri Feb 12 01:45:40 on console
Restored session: Sex 12 Feb 2021 01:39:01 -03
```

Modelo de Transferência de Dados - Datajud

3

E o log abaixo também mostra que o container do MySQL subiu sem erros:

```

rn-vl-mysql is up-to-date
Starting validador_app_1 ... done
Attaching to rn-vl-mysql, validador_app_1
rn-vl-mysql | [Entrypoint] MySQL Docker Image 5.7.33-1.1.19 interactive shell is now zsh.
rn-vl-mysql | [Entrypoint] Initializing database
rn-vl-mysql | [Entrypoint] Database initialized
rn-vl-mysql | Warning: Unable to load '/usr/share/zoneinfo/iso3166.tab' as time zone. Skipping it.
rn-vl-mysql | Warning: Unable to load '/usr/share/zoneinfo/leapseconds' as time zone. Skipping it.
rn-vl-mysql | Warning: Unable to load '/usr/share/zoneinfo/tzdata.zi' as time zone. Skipping it.
rn-vl-mysql | Warning: Unable to load '/usr/share/zoneinfo/zone.tab' as time zone. Skipping it.
rn-vl-mysql | Warning: Unable to load '/usr/share/zoneinfo/zone1970.tab' as time zone. Skipping it.
rn-vl-mysql | [Entrypoint] running /docker-entrpoint-initdb.d/PRD-BNPR-02-02-2021.sql
rn-vl-mysql | [Entrypoint] running /docker-entrpoint-initdb.d/PRD-CORPORATIVO-02-02-2021.sql
rn-vl-mysql | [Entrypoint] running /docker-entrpoint-initdb.d/PRD-SGT_CONSULTA-02-02-2021.sql
rn-vl-mysql | [Entrypoint] Server shut down
rn-vl-mysql | [Entrypoint] MySQL init process done. Ready for start up.
rn-vl-mysql | [Entrypoint] Starting MySQL 5.7.33-1.1.19

```

Para se certificar de que os containers estão executando adequadamente, podemos executar o comando **docker container ls** - verifique os 2 containers (validador_app_1 e rn-vl-mysql), assim como as portas onde eles estão executando:

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
32ad38cf6398	validador_app	"java -XX:+UnlockExp..."	4 minutes ago	Up 2 minutes	0.0.0.0:8081->8081/tcp	validador_app_1
fd3652550379	mysql/mysql-server:5.7	"/entrypoint.sh mysql..."	4 minutes ago	Up 4 minutes (healthy)	3306/tcp, 33060/tcp	rn-vl-mysql

Por padrão, o *backend REST* subirá na porta 8081, e o *MySQL* subirá na porta 3306. Caso não seja possível usar essas portas no seu ambiente operacional, basta alterar o arquivo **docker-compose.yaml** nos pontos marcados pelas setas vermelhas:

Modelo de Transferência de Dados - Datajud

4

```
1 version: '3.7'
2 services:
3
4   rn-mysql:
5     container_name: rn-mysql
6     image: mysql/mysql-server:5.7
7     environment:
8       MYSQL_ROOT_USER: root
9       MYSQL_ROOT_PASSWORD: 12345
10      MYSQL_ROOT_HOST: '%'
11      MYSQL_DATABASE: corporativo
12     expose:
13       - 3306
14     volumes:
15       - ./sql_create:/docker-entrypoint-initdb.d
16     restart: always
17     networks:
18       - banco
19     healthcheck:
20       test: "/usr/bin/mysql --user=root --password=12345 --execute \"SHOW DATABASES;\""
21       interval: 2s
22       timeout: 20s
23       retries: 10
24
25   app:
26     restart: always
27     build:
28       context: ./
29       dockerfile: Dockerfile
30     working_dir: /app
31     expose:
32       - "8080"
33     ports:
34       - "8080:8080"
35     networks:
36       - banco
37     links:
38       - rn-mysql
```

A diretiva **ports** faz um “port forwarding”, e dessa forma realiza mapeamento de uma porta externa com uma porta interna do container.

Após alterar para a porta desejada, basta salvar o arquivo e executar novamente o comando ***docker-compose up***.

A interrupção do funcionamento dos containers pode ser feita rodando o comando: ***docker-compose down***. Todo comando ***docker-compose*** deve ser executado no mesmo diretório onde está localizado o arquivo YAML (***docker-compose.yaml***)

Para conferir se as imagens foram geradas, o comando ***docker image ls***. Resultado da execução do comando abaixo:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
validador_app	latest	865666378dd5	9 hours ago	460MB

Modelo de Transferência de Dados - Datajud

5

Veja que a coluna IMAGE ID traz uma informação importante, que é a identificação da imagem gerada. Ao executar o comando ***docker image rm 865666378dd5***, a imagem será apagada (comando remove - RM). Para forçar o apagamento, situação possível de ocorrer quando foram feitos builds sucessivos de versões da mesma imagem, basta incluir o parâmetro -f: ***docker image rm -f 865666378dd5***

Não é necessário especificar o identificador completo, mas apenas os primeiros caracteres, suficientes para o identificar unicamente. Exemplo: ***docker image rm -f 865***

Alguns comandos úteis para manipulação de imagens Docker:

- * ***docker container stop [container ID]*** - Usado para parar um container
- * ***docker container start [container ID]*** - Usado para iniciar um container
- * ***docker container prune*** - Apaga todos os containers.
- * ***docker image prune*** - Apaga todas as imagens.

Caso o usuário não queira utilizar o banco de dados MySQL como um contêiner (da forma como apresentado acima), é possível fazer a instalação do banco em outro servidor, e apontar a aplicação para o banco novo. Basta passar os parâmetros em linha de comando, como abaixo:

```
java -Dspring.datasource.url=jdbc:mysql://10.203.2.25:3306/corporativo  
-Dspring.datasource.username=root -Dspring.datasource.password=125 -jar  
build/libs/validador-1.0.0.16.jar
```

Os parâmetros acima apontam a aplicação para o novo servidor MySQL que roda na porta 3306 do IP 10.203.2.25. A criação do banco MySQL necessita que sejam importados com os scripts fornecidos na distribuição do Validador (modulo-dados-valida), que ficam no diretório ***sql_create***:

▼  ***sql_create***



PRD-BNPR-02-02-2021.sql



PRD-CORPORATIVO-02-02-2021.sql



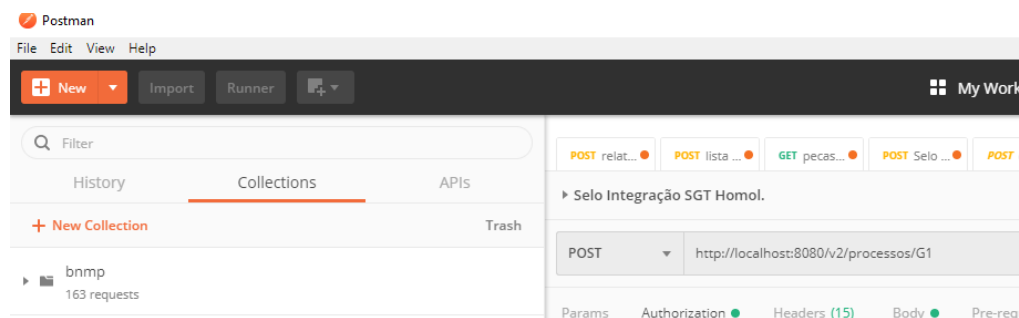
PRD-SGT_CONSULTA-02-02-2021.sql

3. Uso do Validador

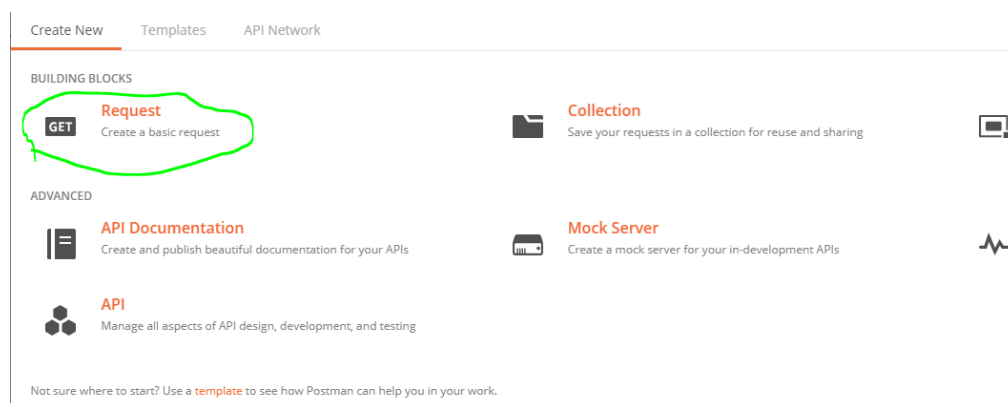
O validador roda como um serviço na máquina, na porta 8081. Pra simular o uso dessa aplicação, faremos uso do Postman (<https://www.postman.com/>). Outros aplicativos para teste em HTTP REST podem ser usados, como o HTTPie (<http://httpie.io>), Insomnia (<https://insomnia.rest/>) ou Nightingale REST Client.

3.1. Fazendo uma Requisição – Simulando um Envio de XML

Ao entrar no Postman, clique no botão “New”, como mostra a tela a seguir.



Depois, clique em “Request”:



Dê um nome para a requisição:

Modelo de Transferência de Dados - Datajud

7

SAVE REQUEST

Requests in Postman are saved in collections (a group of requests).
[Learn more about creating collections](#)

Request name

Request description (Optional)

Make things easier for your teammates with a complete request description.

Descriptions support [Markdown](#)

Na próxima tela, escolha o método HTTP como POST, e a URL, como está apresentado a seguir (supondo que você esteja rodando o Postman na mesma máquina que roda o Validador):

Teste Valida XML Comments (0) Exam

POST ▼ http://localhost:8080/v2/processos/G1 Send ▼ S

Params Authorization Headers Body Pre-request Script Tests Settings Coo

KEY	VALUE	DESCRIPTION	...
Key	Value	Description	

Response



Lembrando que o parâmetro G1, que representa o Grau, pode ser substituído e assumir os valores seguintes:

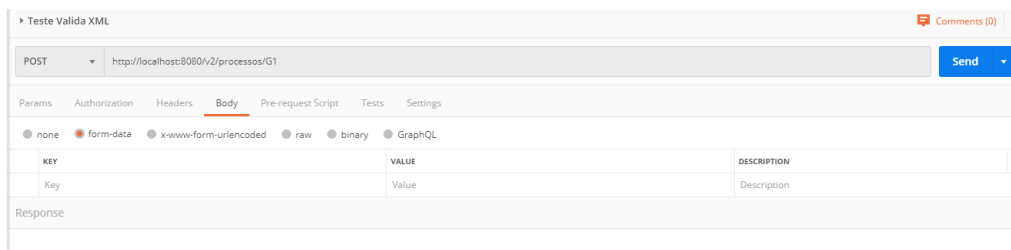
- **SUP** para Tribunais Superiores;
- **G2** para 2º grau;
- **G1** para 1º grau (justiça comum);

Modelo de Transferência de Dados - Datajud

8

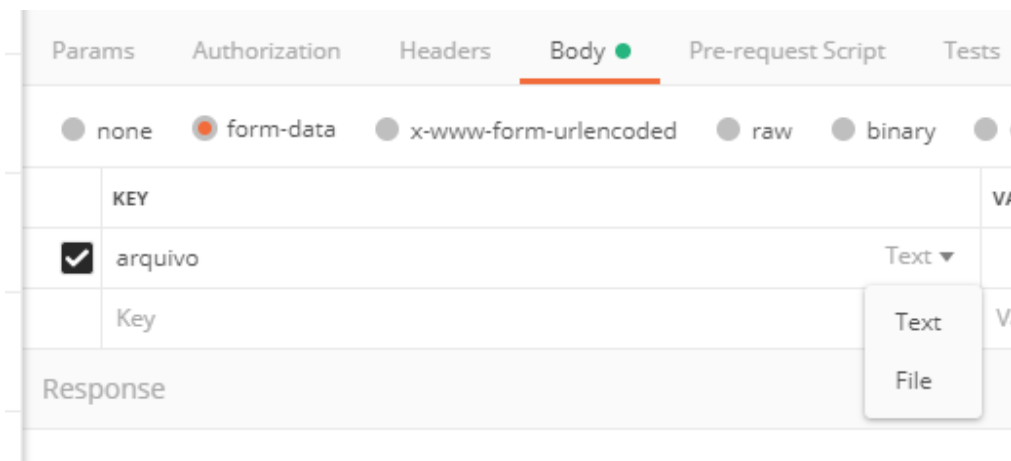
- **TR** para Turmas Recursais;
- **JE** para Juizados especiais;
- **TRU** para Turmas Regionais de Uniformização;
- **TNU** para Turmas Nacionais de Uniformização.

Vá para a aba “Body” e selecione o modo “form-data”:



A interface do Postman mostra a aba "Body" selecionada. O método da requisição é "POST" e o URL é "http://localhost:8080/v2/processos/G1". No topo da aba "Body", há uma barra de seleção com as opções: none, form-data (selecionada), x-www-form-urlencoded, raw, binary e GraphQL. Abaixo, há uma tabela com as seguintes colunas: KEY, VALUE e DESCRIPTION. A primeira linha da tabela contém o texto "Key" na coluna KEY e "Value" na coluna VALUE. A segunda linha contém "Key" na coluna KEY e "Description" na coluna DESCRIPTION. No canto inferior esquerdo da aba, há um campo rotulado "Response".

Depois, acrescente um campo para carregar o arquivo XML. Para isso, será necessário dar um nome qualquer a essa chave (key) e selecionar o tipo do dado, que nesse caso deverá ser “File”:

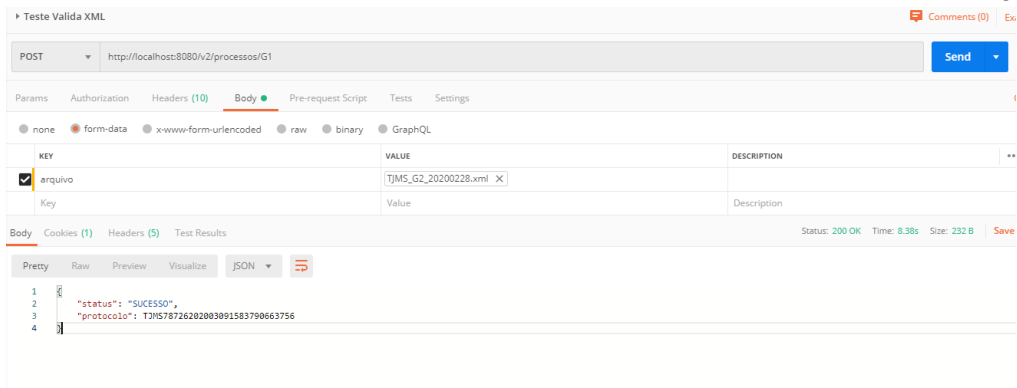


A interface do Postman mostra a aba "Body" selecionada. O método da requisição é "POST" e o URL é "http://localhost:8080/v2/processos/G1". No topo da aba "Body", há uma barra de seleção com as opções: none, form-data (selecionada), x-www-form-urlencoded, raw, binary e GraphQL. Abaixo, há uma tabela com as seguintes colunas: KEY, VALUE e DESCRIPTION. A primeira linha da tabela contém o texto "Key" na coluna KEY e "Value" na coluna VALUE. A segunda linha contém "Key" na coluna KEY e "Description" na coluna DESCRIPTION. No canto inferior esquerdo da aba, há um campo rotulado "Response".

Selecione o arquivo na coluna “Value” (o arquivo, obviamente, deve estar aderente ao XSD do Modelo de Transferência de Dados) e depois clique no botão “Send”, conforme figura abaixo.

Modelo de Transferência de Dados - Datajud

9



Teste Valida XML

POST http://localhost:8080/v2/processos/G1

Params Authorization Headers (10) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION
arquivo	TJMS_G2_20200228.xml	
Key	Value	Description

Body Cookies (1) Headers (5) Test Results

Status: 200 OK Time: 8.38s Size: 232 B Save

Pretty Raw Preview Visualize JSON

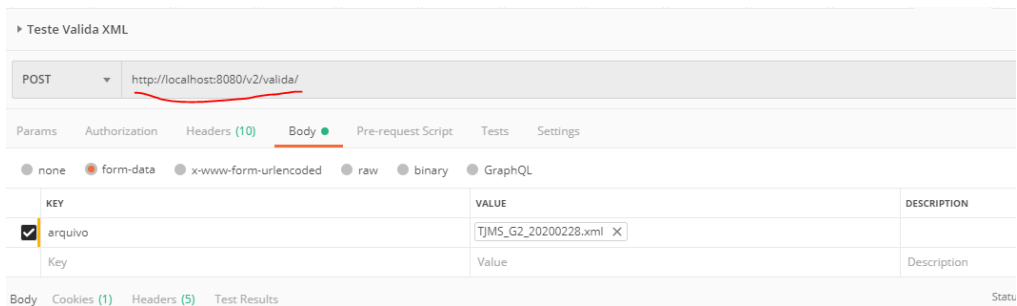
```
1 {
2   "status": "SUCESSO",
3   "protocolo": TJMS78726202003091583790663756
4 }
```

O mesmo resultado seria obtido caso a requisição apontasse para a solução em produção, uma vez que elas implementam o *endpoint* da mesma maneira.

3.2. Operação de Validação

Com essa operação, é possível simular um envio e obter como resultado um conteúdo JSON com o XML processado, incluindo as validações com os dados tabelados em banco e os campos acrescentados por meio do que chamamos de “Rotinas de enriquecimento de dados”. Esse mesmo processo de enriquecimento é usado na solução do CNJ no processamento dos dados XML, e esses campos servem para diversos fins, como calcular a quantidade de movimentos com códigos não aderentes às TPUs (Tabelas Processuais Unificadas), por exemplo.

O endpoint <http://localhost:8081/v2/valida> ou <http://localhost:8081/v1/valida> irá realizar essa transformação e retornar o status de processamento do arquivo XML enviado:



Teste Valida XML

POST http://localhost:8080/v2/valida/

Params Authorization Headers (10) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

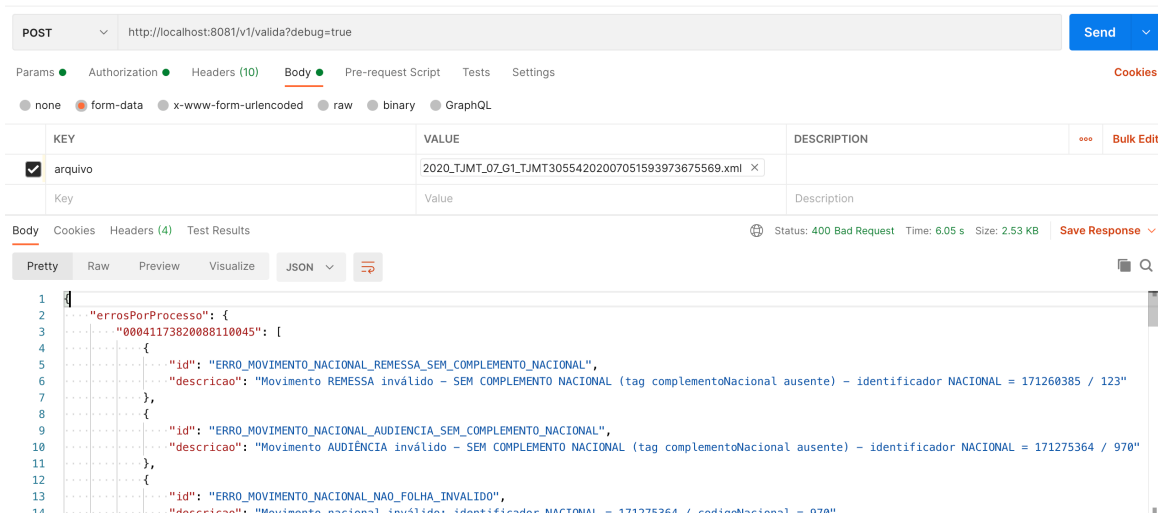
KEY	VALUE	DESCRIPTION
arquivo	TJMS_G2_20200228.xml	
Key	Value	Description

Body Cookies (1) Headers (5) Test Results

Modelo de Transferência de Dados - Datajud

10

Ao clicar em “**Send**”, o resultado será:



POST http://localhost:8081/v1/valida?debug=true

Params Authorization Headers (10) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL

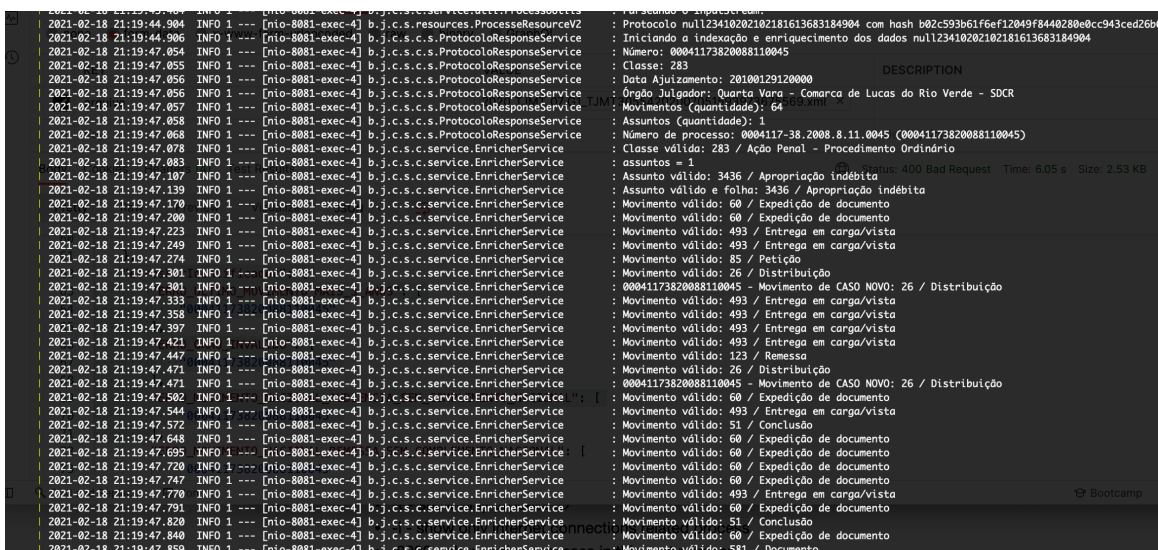
KEY	VALUE	DESCRIPTION
arquivo	2020_TJMT_07_G1_TJMT30554202007051593973675569.xml	
Key	Value	Description

Body Cookies Headers (4) Test Results Status: 400 Bad Request Time: 6.05 s Size: 2.53 KB Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "errosPorProcesso": {
3     "00041173820088110045": [
4       {
5         "id": "ERRO_MOVIMENTO_NACIONAL_REMESSA_SEM_COMPLEMENTO_NACIONAL",
6         "descricao": "Movimento REMESSA inválido - SEM COMPLEMENTO NACIONAL (tag complementoNacional ausente) - identificador NACIONAL = 171260385 / 123"
7       },
8       {
9         "id": "ERRO_MOVIMENTO_NACIONAL_AUDIENCIA_SEM_COMPLEMENTO_NACIONAL",
10        "descricao": "Movimento AUDIÊNCIA inválido - SEM COMPLEMENTO NACIONAL (tag complementoNacional ausente) - identificador NACIONAL = 171275364 / 970"
11      },
12      {
13        "id": "ERRO_MOVIMENTO_NACIONAL_NAO_FOLHA_INVALIDO",
14        "descricao": "Movimento nacional inválido - identificador NACIONAL = 171275364 / complementoNacional = 070"
15      }
16    ]
17  }
18 }
```

Observe que o parâmetro debug foi setado para true (veja a URL da requisição), o que permitirá visualizar as análises feitas pelo Validador na saída do console do seu terminal:



```
2021-02-18 21:19:44.984 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProcessResourceV2 : Protocolo null23410202102181613683184904 com hash b02c593b61f6ef12049f8440280e0cc943ced26b
2021-02-18 21:19:44.986 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Iniciando a indexação e enriquecimento dos dados null23410202102181613683184904
2021-02-18 21:19:44.988 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Número: 00041173820088110045
2021-02-18 21:19:44.990 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Classe: 283
2021-02-18 21:19:44.992 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Data Ajuizamento: 20100129120000
2021-02-18 21:19:44.994 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Órgão Julgador: Quarta Vara - Comarca de Lucas do Rio Verde - SDCR
2021-02-18 21:19:44.996 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Movimentos (quantidade): 64
2021-02-18 21:19:44.998 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Assuntos (quantidade): 1
2021-02-18 21:19:45.000 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.ProtocoloResponseService : Número de processo: 0004117-38.2008.8.11.0045 (00041173820088110045)
2021-02-18 21:19:45.002 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Classe válida: 283 / Ação Penal - Procedimento Ordinário
2021-02-18 21:19:45.004 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Assunto válido: 3436 / Apropriação indébita
2021-02-18 21:19:45.006 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Assunto válido e folha: 3436 / Apropriação indébita
2021-02-18 21:19:45.008 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.010 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.012 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.014 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.016 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 85 / Petição
2021-02-18 21:19:45.018 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 26 / Distribuição
2021-02-18 21:19:45.020 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : 00041173820088110045 - Movimento de CASO NOVO: 26 / Distribuição
2021-02-18 21:19:45.022 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.024 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.026 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.028 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 123 / Remessa
2021-02-18 21:19:45.030 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 26 / Distribuição
2021-02-18 21:19:45.032 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : 00041173820088110045 - Movimento de CASO NOVO: 26 / Distribuição
2021-02-18 21:19:45.034 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.036 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.038 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 51 / Conclusão
2021-02-18 21:19:45.040 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.042 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.044 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.046 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.048 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 493 / Entrega em carga/vista
2021-02-18 21:19:45.050 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.052 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 51 / Conclusão
2021-02-18 21:19:45.054 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 60 / Expedição de documento
2021-02-18 21:19:45.056 INFO 1 --- [nio-8081-exec-4] b.j.c.s.c.service.EnricherService : Movimento válido: 51 / Documento
```

Modelo de Transferência de Dados - Datajud

11

Veja também que o código de status HTTP da resposta (HTTP Response) será o 400, pois o arquivo enviado contém erros de processamento. Os erros de processamento podem ser verificados através de 2 estruturas de array no JSON de resposta - `errosPorProcesso` e `errosPorIdentificador`.

A estrutura `errosPorProcesso`, como o nome sugere, é uma lista de todos os erros, agrupados por processo. Ou seja, num primeiro nível eu tenho o número do processo, e dentro de cada processo eu tenho a lista de erros identificados nesse processo:

```
"errosPorProcesso": {  
  "00041173820088110045": [  
    {  
      "id": "ERRO_MOVIMENTO_NACIONAL_REMESSA_SEM_COMPLEMENTO_NACIONAL",  
      "descricao": "Movimento REMESSA inválido - SEM COMPLEMENTO NACIONAL (tag  
complementoNacional ausente) - identificador NACIONAL = 171260385 / 123"  
    },  
    {  
      "id": "ERRO_MOVIMENTO_NACIONAL_AUDIENCIA_SEM_COMPLEMENTO_NACIONAL",  
      "descricao": "Movimento AUDIÊNCIA inválido - SEM COMPLEMENTO NACIONAL  
(tag complementoNacional ausente) - identificador NACIONAL = 171275364 / 970"  
    }  
  ]  
}
```

Já a estrutura `errosPorIdentificador` é o inverso da estrutura anterior, ou seja, é uma lista de todos os processos, agrupados por erro. Ou seja, num primeiro nível eu tenho o identificador do erro, e dentro de cada erro eu tenho a lista de processos que apresentou ao menos 1 incidência desse tipo de erro:

```
"errosPorIdentificador": {  
  "ERRO_ULTIMO_MOVIMENTO MAIS_3_ANOS": [  
    "00041173820088110045"  
  ],  
  "ERRO_GRAU_INVALIDO": [  

```

Modelo de Transferência de Dados - Datajud

12

```
"00041173820088110045"  
],  
"ERRO_MOVIMENTO_NACIONAL_AUDIENCIA_SEM_COMPLEMENTO_NACIONAL": [  
    "00041173820088110045"  
]
```

Existem 26 identificadores de erros. Eles são auto-descritivos, e vem acompanhado (na estrutura `errosPorProcesso`) de um campo descrição que informa a origem da ocorrência no documento XML analisado:

1. *ERRO_MOVIMENTO_LOCAL_INVALIDO*

Código CNJ do movimento local (atributo `codigoPaiNacional`) não corresponde a código válido ou não folha (movimento em nível não folha é considerado folha, caso todos filhos estejam inativados);

2. *ERRO_MOVIMENTO_NACIONAL_INVALIDO*

Código CNJ do movimento nacional (atributo `codigoNacional`) não bate com a TPU (Res. 46 CNJ) ou não está em último nível;

3. *ERRO_ULTIMO_MOVIMENTO MAIS 3 ANOS*

Tempo desde última movimentação acima de 3 anos desde que não esteja arquivado ou suspenso (desconsiderar movimentos 246, 22, 265, 11975);

4. *ERRO_ASSUNTO_LOCAL_INVALIDO*

Código CNJ do movimento local (atributo `codigoPaiNacional`) não corresponde a código válido ou não folha (movimento em nível não folha é considerado folha, caso todos filhos estejam inativados);

5. *ERRO_ASSUNTO_NACIONAL_INVALIDO*

Modelo de Transferência de Dados - Datajud

13

Código CNJ do movimento nacional (atributo `codigoNacional`) não bate com a TPU (Res. 46 CNJ) ou não está em último nível;

6. ERRO_ASSUNTO_NAO_FOLHA_INVALIDO

Assunto genérico, ou seja, assuntos que não estão no nível folha;

7. ERRO_SEM_ASSUNTOS

Lista de assuntos vazia (ou seja, sem assuntos);

8. ERRO_MOVIMENTO_COD_PAI_NACIONAL_INVALIDO

Atributo `codigoPaiNacional` não foi preenchido no movimento local;

9. ERRO_MOVIMENTO_NACIONAL_AUDIENCIA_SEM_COMPLEMENTO_NACIONAL

Movimento nacional de audiência sem complemento nacional (tag `complementoNacional` do movimentos) (movimento 82, 123).

10.ERRO_MOVIMENTO_LOCAL_AUDIENCIA_SEM_COMPLEMENTO_NACIONAL

Movimento local de audiência sem complemento nacional (tag `complementoNacional` do movimentos) (movimento 82, 123).

11. ERRO_MOVIMENTO_NACIONAL_REMESSA_SEM_COMPLEMENTO_NACIONAL

Movimento nacional de remessa sem complemento nacional (tag `complementoNacional` do movimento) (movimento 970, 12739, 12740, 12741, 12742, 12743, 12744, 12745, 12746, 12747, 12749, 12750, 12751, 12753, 12624, 14096);

12.ERRO_MOVIMENTO_LOCAL_REMESSA_SEM_COMPLEMENTO_NACIONAL

Modelo de Transferência de Dados - Datajud

14

Movimento local de remessa sem complemento nacional (tag complementoNacional do movimento) (movimento 970, 12739, 12740, 12741, 12742, 12743, 12744, 12745, 12746, 12747, 12749, 12750, 12751, 12753, 12624, 14096);

13.ERRO_MOVIMENTO_NACIONAL_NAO_FOLHA_INVALIDO

Código CNJ do movimento nacional (atributo codigoNacional) bate com a TPU (Res. 46 CNJ), mas não esta em último nível;

14.ERRO_MOVIMENTO_LOCAL_NAO_FOLHA_INVALIDO

Código CNJ do movimento local (atributo codigoNacional) bate com a TPU (Res. 46 CNJ), mas não esta em último nível;

15.ERRO_ULTIMO_MOVIMENTO_DATA_INVALIDA

Último movimento (ou seja, movimento mais novo) está com a data inválida.

16.ERRO_SEM_MOVIMENTO_CASO_NOVO

Não possui movimento de caso novo (movimentos 26, 132, 981, 391, 393);

17.ERRO_CLASSE_PROCESSUAL_INVALIDA

Código CNJ da classe não bate com a TPU (Res. 46 CNJ). Atributo classeProcessual da tag dadosBasicos tem conteúdo inválido;

18.ERRO_ORGAO_JULGADOR_INVALIDO

Código do órgão não bate com os órgãos oficiais (Anexo II Res. 76 CNJ);

19.ERRO_SEM_NOME_PARTE

Processo sem nome no polo ativo ou passivo (atributo nome da tag pessoa);

20.ERRO_PESSOA_SEM_DOCUMENTO_PRINCIPAL

Processo com parte sem o atributo numeroDocumentoPrincipal da tag pessoa;

21.ERRO_PESSOA_SEM_DOCUMENTO

Processo com parte sem documento (ausência da tag documento na tag pessoa);

22.ERRO_PESSOA_COM_DOCUMENTO_CMF_SEM_NUMERO_DOCUMENTO_PRINCIPAL

Processo com parte com documento do tipo CMF (CPF ou CNPJ), mas sem a tag numeroDocumentoPrincipal;

23.ERRO_PROCESSO_SEM_POLO_ATIVO

Processo sem parte no polo ativo;

24.ERRO_GRAU_INVALIDO

Processo informação de grau (sem atributo grau em dadosBasicos, e sem grau no nível raiz);

25.ERRO_NUMERO_PROCESSO_JTR_INVALIDO

Validação do segmento de justiça do número de processo inválido (código JTR não faz parte do conjunto de códigos JTR válidos);

26.ERRO_NUMERO_PROCESSO

Número de processo com dígito verificador inválido. Campo DV inválido no modelo de numeração única (DDDDDDD-DV.AAAA.J.TR.OOOO).

27.ERRO_ORGAO_JULGADOR_INVALIDO_NAO_CORRESPONDE_SIGLA_TRIBUNAL

Atributo `codigoOrgao` da tag `orgaoJulgador` do processo não pertence ao Tribunal especificado na tag `siglaTribunal`.

3.3.Outros Endpoints

3.3.1. `/processos/json`

Método HTTP POST. Processa o XML e retorna uma representação do XML em JSON, tal como se o XML tivesse sido convertido para ser representado num índice ElasticSearch.

3.3.2. `/processos/json/{grauJurisdicao}`

Método HTTP POST. Processa o XML e retorna uma representação do XML em JSON, tal como se o XML tivesse sido convertido para ser representado num índice ElasticSearch. Permite passar um parâmetro com o grau da jurisdição: G1, G2, SUP, etc.

3.3.3. `/processos/{grauJurisdicao}`

Método HTTP POST. Simula um envio do XML para o módulo de recepção de dados. Retorna como resultado a mesma estrutura JSON que é respondida com a execução do endpoint para envio de arquivos em produção:

```
{  
  "status": "SUCESSO",  
  "protocolo": TJRJ24641202103081615256266778  
}
```

Modelo de Transferência de Dados - Datajud

17

O status SUCESSO é lançado com o simples envio do arquivo, da mesma forma que ocorre com o serviço em produção. Ou seja, o status SUCESSO não significa que o XML foi processado com sucesso, e sim que foi recebido com sucesso.

De toda forma, o arquivo XML é validado, e o resultado do processamento pode ser verificado na saída do log do contêiner:

```
nio-8081-exec-2] b.j.c.s.c.s.ProtocoloResponseService : 27: ERRO_MOVIMENTO_NACIONAL_INVALIDO, Movimento local inválido: identificador NA
nio-8081-exec-2] b.j.c.s.c.s.ProtocoloResponseService : 28: ERRO_MOVIMENTO_NACIONAL_INVALIDO, Movimento local inválido: identificador NA
nio-8081-exec-2] b.j.c.s.c.s.ProtocoloResponseService : 29: ERRO_MOVIMENTO_NACIONAL_INVALIDO, Movimento local inválido: identificador NA
nio-8081-exec-2] b.j.c.s.c.s.ProtocoloResponseService : 30: ERRO_MOVIMENTO_NACIONAL_INVALIDO, Movimento local inválido: identificador NA
nio-8081-exec-2] b.j.c.s.c.s.ProtocoloResponseService : 31: ERRO_MOVIMENTO_NACIONAL_INVALIDO, Movimento local inválido: identificador NA
nio-8081-exec-6] b.j.c.s.c.s.service.util.ProcessoUtils : Parseando o InputStream.
nio-8081-exec-6] b.j.c.s.c.s.resources.ProcessResourceV2 : Protocolo TJRJ24641202103081615256266778 com hash 80d146d36f381fc51b2077cfc3817f
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Iniciando a indexação e enriquecimento dos dados TJRJ24641202103081615256266778
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Número: 00233992220208190206
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Classe: 7
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Data Ajuizamento: 20201117000000
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Órgão Julgador: SANTA CRUZ REGIONAL 1 VARA CIVEL
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Movimentos (quantidade): 2
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Assuntos (quantidade): 1
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : Número de processo: 0023399-22.2020.8.19.0206 (00233992220208190206)
nio-8081-exec-6] b.j.c.s.c.s.service.EnricherService : Classe válida: 7 / Procedimento Comum Cível
nio-8081-exec-6] b.j.c.s.c.s.service.EnricherService : assuntos = 1
nio-8081-exec-6] b.j.c.s.c.s.service.EnricherService : Assunto válido: 8960 / Processo e Procedimento
nio-8081-exec-6] b.j.c.s.c.s.service.EnricherService : Assunto válido e NÃO folha: 8960 / Processo e Procedimento
nio-8081-exec-6] b.j.c.s.c.s.service.EnricherService : Movimento válido: 11010 / Mero expediente
nio-8081-exec-6] b.j.c.s.c.s.service.EnricherService : Data atual: 08/03/2021
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : qtdErro = 1
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : ERRO - Processo 00233992220208190206 NÃO foi validado, pois apresenta erros!
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : 1 qtdLote
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : tempo gasto: 1 segundos
nio-8081-exec-6] b.j.c.s.c.s.ProtocoloResponseService : RESUMO do processamento - Protocolo TJRJ24641202103081615256266778
```

3.3.4. /versão

Método HTTP GET. Retorna a versão da API:

GET

http://localhost:8081/v1/versao

Params

Authorization

Headers (10)

Body

Pre-request Script

Tests

Query Params

KEY	VALUE
Key	Value

Body

Cookies

Headers (5)

Test Results

Pretty

Raw

Preview

Visualize

Text

1 Versão: 1.0.0.16

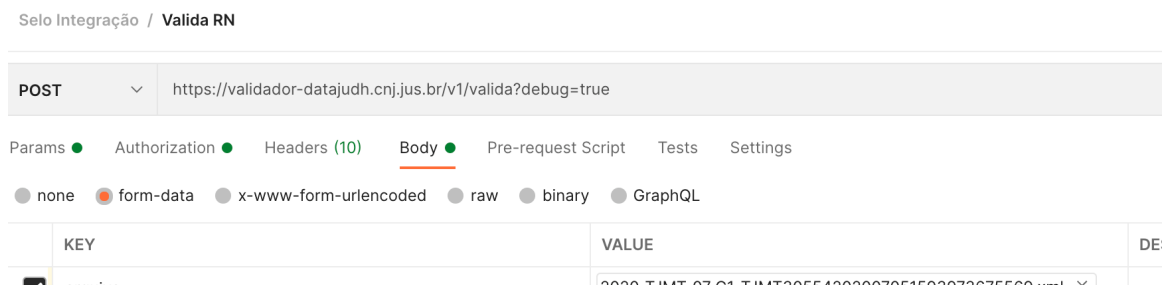
3.3.5./validaXSD

Similar à operação **/valida**, mas nesse caso, são levados em conta apenas os critérios mínimos de aceitação do processo: número de processo conforme a Resolução 65 (numeração única), classe processual válida e órgão julgador válido.

3.4. Instância do Validador Hospedado no CNJ

Há uma versão do Validador hospedado no CNJ, para permitir o uso por parte dos tribunais. O caminho desse Validador é: <https://validador-datajudh.cnpjus.br/>

A configuração no Postman ficaria da forma seguinte:



O uso dessa apresentação do Validador hospedado no CNJ tem algumas desvantagens consideráveis, sendo uma delas a impossibilidade de analisar os erros de log, função permitida através do parâmetro debug=true.

4. Acesso ao código-fonte do projeto Validador

Modelo de Transferência de Dados - Datajud

19

O código-fonte do sistema Validador está disponível no repositório GIT do CNJ, no grupo GIT-JUS.

O caminho do Projeto do Validador no Git.Jus: <https://git.cnj.jus.br/git-jus/datajud/validador>

Caso não tenha acesso, basta seguir as orientações em: <https://www.cnj.jus.br/sistemas/git-jus/>