

Sumário

1. O que é a aplicação ArqSIGN	3
2. URL da API ArqSIGN	3
3. Requisitos para uso da API.....	3
4. Limites de Requisições da API.....	4
5. Métodos disponíveis na API.....	4
6. Fluxo ideal considerando alguns métodos disponíveis.....	6
7. Detalhes dos métodos.....	6
7.1 POST /api/v1/processo/enviar-documento-para-assinar.....	6
7.1.1 Requisição.....	6
7.1.2 Retorno.....	15
7.2 GET /api/v1/processo/{idprocesso}.....	16
7.2.1 Requisição.....	16
7.2.2 Retorno.....	16
7.2.3 Exemplo de JSON de Retorno.....	17
7.3 GET /api/v1/processo/{idProcesso}/status-do-processo.....	18
7.3.1 Requisição.....	18
7.3.2 Retorno.....	18
7.4 GET /api/v1/processo/{idprocesso}/dados-signatarios.....	19
7.4.1 Requisição.....	19
7.4.2 Retorno.....	20
7.4.3 Exemplo de JSON de Retorno.....	20
7.5 PATCH /api/v1/processo/{idProcesso}/reenviar-processo.....	21
7.5.1 Requisição.....	21
7.5.2 Retorno.....	23
7.6 PATCH /api/v1/processo/{idProcesso}/cancelar-processo.....	23
7.6.1 Requisição.....	23
7.6.2 Retorno.....	24

Controle de Versão

Versão do Manual	Data	Descrição das Alterações
01.00	05/09/2022	Primeira Versão
01.01	21/09/2022	Ajuste em: Tópico 7.1.1 – Requisição (Ref. 10.13 – dados de posição dos parâmetros de “definirPosicaoAssinaturaManual”)
01.02	07/10/2022	Inclusão do: Tópico 4 - Limites de Requisições da API
01.03	25/10/2022	Ajuste em: Tópico 3 - Requisitos para uso da API Ajuste em: Tópico 5 - Métodos disponíveis na API (Inclusão do 1º parágrafo)
01.04	01/03/2023	Ajuste em: Tópico 5 - Métodos disponíveis na API Incluídos os métodos: <ul style="list-style-type: none">• Reenviar processo• Buscar dados dos signatários• Cancelar processo

1. O que é a aplicação ArqSIGN

A **ArqSIGN** é uma plataforma online com o objetivo de assinar documentos de forma eletrônica usando o certificado digital da plataforma ou o seu certificado digital ICP Brasil.

A API da **ArqSIGN** vem possibilitar a Você enviar documentos do seu sistema para serem assinados via plataforma **ArqSIGN**

2. URL da API ArqSIGN

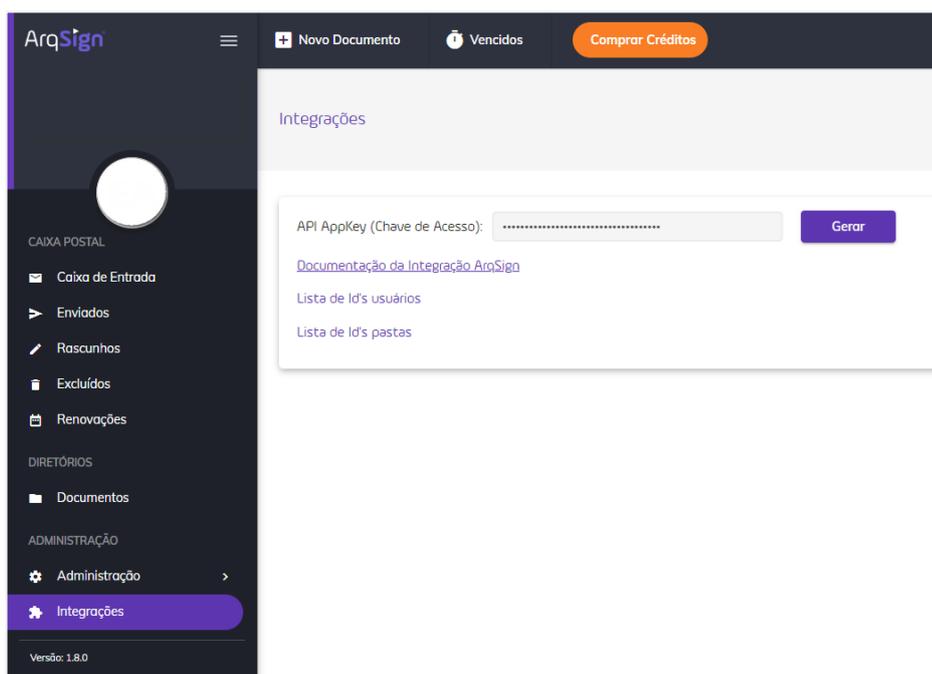
A URL com a documentação online de API é: <https://api.arqsign.com/index.html>

3. Requisitos para uso da API

- Ter uma conta na plataforma **ArqSIGN** ativa;
- Ter uma chave de acesso a API válida, esta chave chamamos de **AppKey**;
- Para o método **POST/api/v1/processo/enviar-documento-para-assinar**, será necessário ter em mãos:
 - **ID de um usuário da conta com status ativo**: para ser responsável pelos documentos que serão enviados pela API;
 - **ID de pasta**: onde o documento a ser gerado deve ser armazenado na plataforma.

Nota: A funcionalidade de Diretórios irá permitir ao Cliente ter várias pastas organizadoras dentro da plataforma, esta funcionalidade será liberada em breve para os planos pagos. Enquanto a funcionalidade de Diretórios não estiver liberada, ou caso o seu plano seja grátis, a sua conta terá somente uma pasta que chamamos de “Pasta Raiz” e neste caso, você precisará deste ID para usar a API.

Para obter a sua AppKey, os ID’s dos usuários e os ID’s das Pastas, acesse a sua conta na aplicação ArqSIGN (app.arqsign.com) e clique no menu “Integrações” conforme mostra a figura abaixo:



Nesta tela temos as opções:

- API AppKey (Chave de Acesso) - Gerar: Ao clicar no botão gerar a aplicação irá criar ou substituir a chave de acesso a API. Se a conta não possuir uma chave, então uma será criada, se a aplicação já possuir uma chave, então esta será substituída. A chave será exibida para o usuário copiar somente uma vez, caso esta cópia não seja realizada, uma nova chave precisará ser gerada.
- Documentação da Integração ArqSIGN: Ao clicar neste link, a aplicação irá abrir a página api.arqsign.com com os métodos disponíveis até o momento.
- Lista de Id's usuários: Ao clicar neste link a aplicação irá fazer o download de um arquivo csv com a lista de todos os usuários ativos na conta e seu respectivo ID.
- Lista de Id's pastas: Ao clicar neste link a aplicação irá fazer o download de um arquivo csv com a lista de todas as pastas não excluídas da conta e seu respectivo ID.

4. Limites de Requisições da API

Limite de Requisições da API por Hora e por Mês*		
Planos Mensais	Por hora	Por Mês
Conta Teste Grátis - 15 dias	5	25
Profissional Mensal - 10 envios	10	40
Profissional Mensal - 30 envios	30	120
Profissional Mensal - 60 envios	50	200
Corporativo Mensal - ILIMITADO	50	200
Planos Anuais	Por hora	Por Mês
Profissional Anual - 120 envios	50	200
Profissional Anual - 360 envios	50	200
Profissional Anual - 720 envios	50	200
Corporativo Anual - ILIMITADO	50	200

* Os limites de requisições garantem estabilidade e segurança dos serviços para os clientes.

Entretanto, se o seu negócio tem uma demanda maior do que a mencionada acima, com planejamento, temos condições de atender à sua demanda sem prejudicar a estabilidade e a segurança dos serviços.

Para isso, entre em contato conosco através de 4003-8839 ou comercial@arqsign.com que trabalharemos em um Plano sob medida para você.

5. Métodos disponíveis na API

Todos os métodos que retornam campos de data e hora na API estão no fuso horário UTC+0.

Os métodos de API liberados para uso na plataforma são:

- **POST** [/api/v1/processo/enviar-documento-para-assinar](#)

O objetivo deste método é permitir que você envie documento para ser assinado via plataforma ArqSIGN.

Ver o tópico: 6 - Detalhes dos métodos

- **GET/api/v1/processo/{idprocesso}**

O objetivo deste método é permitir que o Cliente busque os dados completos de um processo, incluindo o documento e o registro de assinatura, caso já exista alguma assinatura no documento.

Para evitar que o método retorne documentos ainda não assinados ou em processo de assinatura, utilize o método de buscar o status do processo para checar se o processo em questão se encontra com o status “Concluído”.

Neste método você irá nos enviar o ID do Processo, e nós retornaremos um JSON completo com as informações do processo.

Ver o tópico: 6 - Detalhes dos métodos

- **GET/api/v1/processo/{idProcesso}/status-do-processo**

O objetivo deste método é permitir que você busque o status de um processo de assinatura, para que evite buscar o processo como um todo pelo método “**GET/api/v1/processo/{idprocesso}**” antes que este esteja com status “Concluído”. Neste método você irá nos enviar o ID do Processo, e nós retornaremos um JSON com o nome e status atual do mesmo.

Ver o tópico: 6 - Detalhes dos métodos

- **GET/api/v1/processo/{idprocesso}/dados-signatarios**

O objetivo deste método é permitir que você busque os dados dos signatários, com ação de Assinar Online, de um processo de assinatura.

Neste método você irá nos enviar o ID do Processo, e nós retornaremos um JSON completo com as informações do processo e dos signatários.

- **PATCH/api/v1/processo/{idProcesso}/reenviar-processo**

O objetivo deste método é permitir que você reenvie o processo para os destinatários pendentes de assinaturas na ordem atual.

Você poderá informar para qual destinatário pendente de assinatura deseja reenviar o documento. Caso não informe os destinatários, o serviço reenvia o processo para todos os destinatários participantes do processo com ação de assinar online e que estejam pendentes de assinaturas na ordem atual.

Além disso, você poderá editar os destinatários pendentes de assinatura que deseja reenviar o processo.

Neste método você irá nos enviar o ID do Processo, e nós reenviaremos para os destinatários pendentes de assinatura na ordem de assinatura atual, conforme dados informados no JSON.

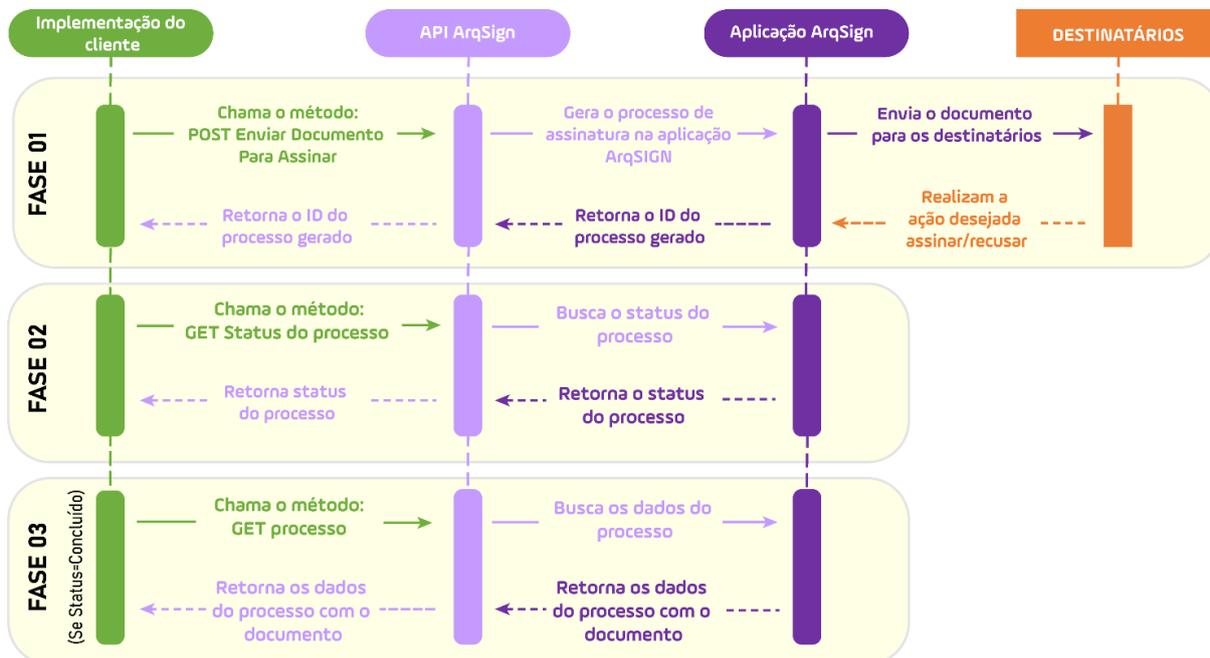
- **PATCH/api/v1/processo/{idProcesso}/cancelar-processo**

O objetivo deste método é permitir que você cancele o processo que esteja em andamento.

Neste método você irá nos enviar o ID do Processo, e nós cancelaremos o processo informado.

6. Fluxo ideal considerando alguns métodos disponíveis

Abaixo mostramos um fluxo ideal para implementação de três métodos dos disponíveis:



- **Fase 01:** Na fase 01 você chama o método `POST/api/v1/processo/enviar-documento-para-assinar` para enviar um documento a ser assinado. Com a resposta de sucesso, a API retornará o ID do processo gerado e você deve guardá-lo para usar este ID como parâmetro nos outros métodos.
- **Fase 02:** Na fase 02 você chama o método `GET/api/v1/processo/{idProcesso}/status-do-processo` para monitorar o status do processo gerado na Fase 01. Aconselhamos chamar este método no máximo 1x ao dia, não é necessário chamá-lo o tempo todo, uma vez que o status de um processo somente será concluído após a assinatura do documento por todos os signatários.
- **Fase 03:** Assim que o processo estiver com o status "Concluído", você pode passar para a fase 03, quando chamará o método `GET/api/v1/processo/{idprocesso}` para obter os dados completos do processo e o arquivo assinado.

7. Detalhes dos métodos

7.1 `POST/api/v1/processo/enviar-documento-para-assinar`

7.1.1 Requisição

Header				
Ref.	Parâmetro	Formato	Valor	Requerido
01	Authorization	AppKey	O AppKey identifica a conta.	Sim
Body				
Ref.	Parâmetro	Formato	Valor	Requerido
03	"ConfiguracaoAvancadas":			
03.01	"tempoExpiracaoDocumentoDias"	"smallint"	Valor mínimo: {1} Valor máximo: {999}	Não

			Ou null para usar o padrão da conta.	
03.02	"avisoAntesExpiracaoDocumentoDias"	"smallint"	Valor mínimo: {1} Valor máximo: {99} Ou null para usar o padrão da conta.	Não
03.03	"frequenciaLembretesDias "	"smallint"	Valor mínimo: {1} Valor máximo: {999} Ou null para usar o padrão da conta.	Não
04	"idPasta"	"guid"	{IdPasta}	Sim
05	"renovacaoMeses"	"tinyint"	Tempo em meses para renovação do documento ou null para não usar renovação.	Não
06	"assinarOrdemDestinatarios"	"bit"	1 = true, 0 = False	Sim
07	"nome"	"varchar(250)"	Nome do processo	Sim
08	"mensagemPadrao":			
08.01	"titulo"	"varchar (150)"	Assunto padrão para todos os destinatários ou null para a aplicação enviar a notificação padrão.	Não
08.02	"texto"	"varchar (max)"	Mensagem padrão para todos os destinatários ou null para a aplicação enviar a notificação padrão.	Não
09	"idResponsavel"	"guid"	Id do usuário responsável pelo documento	Sim
10	"Destinatários"			
10.01	"ordem"	"tinyint"	Valor numérico definindo a ordem de assinatura para o destinatário Ou null para não usar ordem de assinatura.	Não
10.02	"nome"	"varchar (150)"	Nome do destinatário	Sim
10.03	"idTipoAcao"	"tinyint"	1 - Assina Online 2 - Receber uma cópia	Sim
10.04	"idTipoAssinatura"	"tinyint"	1 - Assinatura Eletrônica 3 - Certificado Digital - ICP Brasil 4 - Certificado Digital - Outros 5 - Sem Assinatura	Sim
10.05	"idFormaEnvio"	"bit"	Forma de envio do processo 1 - E-mail 2 - Whatsapp	Sim
10.06	"email"	"varchar(150)"	Obrigatório se idFormaEnvio = 1	Sim
10.07	"telefone"	"varchar(20)"	Obrigatório se idFormaEnvio = 2	Sim
10.08	"Seguranca":			
10.08.01	"idMeioEnvio"	"tinyint"	1- SMS (Somente Brasil) 2 - Whatsapp 3 - Email 4 - Não enviar Ou null não usar o código de segurança.	Não
10.08.02	"codigoSeguranca"	"char(4)"	Código de segurança para acessar o documento. Ou null não usar o código de segurança.	Não
10.08.03	"codigoSegurancaEmail"	"varchar(150)"	Email para enviar o código de segurança. *1 Obrigatório se idMeioEnvio = 3. Ou null não usar o código de segurança.	Não *1
10.08.04	"codigoSegurancaTelefone"	"varchar(20)"	Telefone para enviar o código de segurança.	Não *2

			*2 Obrigatório se idMeioEnvio = 1 ou 2. Ou null não usar o código de segurança.	
10.08.05	"reenviarCodigo"	"bit"	1 = true 0 = False Ou null	Não
10.09	"mensagemPersonalizada":			
10.09.01	"titulo"	"varchar(150)"	Assunto da mensagem personalizada do destinatário, em questão. Ou null para não usar notificação personalizada.	Não
10.09.02	"texto"	"varchar(max)"	Mensagem personalizada do destinatário, em questão. Ou null para não usar notificação personalizada.	Não
10.10	"destinatariosEntradaDto":			
10.10.01	"infComplNomeSignatarioObrigatorio"	"bit"	1 = true 0 = False Ou null quando idTipoAcao = 2	Sim
10.10.02	"infComplDocumentoSignatarioObrigatorio"	"bit"	1 = true 0 = False Ou null quando idTipoAcao = 2	Sim
10.10.03	"infComplRazaoSocialObrigatorio"	"bit"	1 = true 0 = False Ou null quando idTipoAcao = 2	Sim
10.10.04	"infComplIdentificadorObrigatorio"	"bit"	1 = true 0 = False Ou null quando idTipoAcao = 2	Sim
10.10.05	"idInfComplTipoDocumentoSignatarioPF"	"tinyint"	1 = CPF 2 = CNH 3 = RG 4 = Outro Ou null quando idTipoAcao = 2	Sim
10.10.06	"infComplDocumentoNomePF"	"varchar(150)"	Nome do documento de identificação da PF. *1 Obrigatório se idInfComplTipoDocumentoSignatarioPF = 4.	Não *1
10.10.07	"idInfComplTipoDadosIdentificadorPF"	"tinyint"	1 - Texto 2 - Numérico *1 Obrigatório se idInfComplTipoDocumentoSignatarioPF = 4.	Não *1
10.10.08	"infComplQtdeCaracteresPF"	"tinyint"	Quantidade de caracteres para o documento de identificação de PF. *1 Obrigatório se idInfComplTipoDocumentoSignatarioPF = 4.	Não *1
10.10.09	"idInfComplTipoDocumentoSignatarioPJ"	"tinyint"	4 = Outro 5 = CNPJ	Não *1
10.10.10	"infComplDocumentoNomePJ"	"varchar(150)"	Nome do documento de identificação da PJ. *1 Obrigatório se idInfComplTipoDocumentoSignatarioPJ = 4.	Não *1
10.10.11	"idInfComplTipoDadosIdentificadorPJ"	"tinyint"	1 - Texto 2 - Numérico *1 Obrigatório se idInfComplTipoDocumentoSignatarioPJ = 4.	Não *1
10.10.12	"infComplQtdeCaracteresPJ"	"tinyint"	Quantidade de caracteres para o documento de identificação de PJ. *1 Obrigatório se idInfComplTipoDocumentoSignatarioPJ = 4.	Não *1

10.11	"Anexos":			
10.11.01	"anexoDocumentoNome"	"VarChar(150)"	Nome do anexo ou null caso não seja necessário usar anexos.	Não
10.11.02	"anexoObrigatorio"	"bit"	1 = true 0 = False Ou null caso não seja necessário usar anexos.	Não
10.11.03	"anexoExibirTodosDestinatarios"	"bit"	1 = true 0 = False Ou null caso não seja necessário usar anexos.	Não
10.12	"definirPosicaoAssinaturaAutomatica" *1 É obrigatório usar "definirPosicaoAssinaturaAutomatica" ou "definirPosicaoAssinaturaManual"	"bit"	null, 1 – PF 2 – PJ 3 – PF e PJ	Sim/Não *1
10.13	"definirPosicaoAssinaturaManual" *1 É obrigatório usar "definirPosicaoAssinaturaAutomatica" ou "definirPosicaoAssinaturaManual"			
10.13.01	"pessoaFisica"			
10.13.01.01	"Pagina"	"int"	Página do arquivo que será inserido a representação visual da assinatura - PF	Sim/Não
10.13.01.02	"Altura"	"decimal"	Altura da representação visual da assinatura - PF	Sim/Não
10.13.01.03	"Largura"	"decimal"	Largura da representação visual da assinatura - PF	Sim/Não
10.13.01.04	"PosicaoX"	"decimal"	Posição x da representação visual da assinatura - PF	Sim/Não
10.13.01.05	"PosicaoY"	"decimal"	Posição Y da representação visual da assinatura - PF	Sim/Não
10.13.02	"pessoaJuridica"			
10.13.02.01	"Pagina"	"int"	Página do arquivo que será inserido a representação visual da assinatura - PJ	Sim/Não
10.13.02.02	"Altura"	"decimal"	Altura da representação visual da assinatura - PJ	Sim/Não
10.13.02.03	"Largura"	"decimal"	Largura da representação visual da assinatura - PJ	Sim/Não
10.13.02.04	"PosicaoX"	"decimal"	Posição x da representação visual da assinatura - PJ	Sim/Não
10.13.02.05	"PosicaoY"	"decimal"	Posição Y da representação visual da assinatura - PJ	Sim/Não
11	Documento:			
11.01	"arquivo"	"String Base64"		Sim
11.02	"nomeComExtensao"	"String"		Sim

Detalhamento do Header e Body

Geral: Os parâmetros não obrigatórios listados abaixo devem constar no JSON, e caso não seja necessário para o documento, enviar com o valor null na frente:

- Ref. 03 "ConfiguracoesAvancadas"
- Ref. 05 "RenovacoesMeses"
- Ref. 08 "MensagemPadrao"
- Ref. 10.08 "Seguranca"
- Ref. 10.09 "mensagemPersonalizada"
- Ref. 10.10 "destinatariosEntradaDto"
- Ref. 10.11 "Anexos"

→ Ref. 10.12 “definirPosicaoAssinaturaAutomatica”

→ Ref. 10.13 “definirPosicaoAssinaturaManual”

Ref. 01: “AppKey” é a chave de autorização para se autenticar na API. Esta chave deve ser válida e estar vinculada a uma conta ArqSIGN ativa.

Ref. 03: “ConfiguracoesAvancadas” - As configurações avançadas é um valor opcional para enviar um documento a ser assinado. Quando esta configuração não é enviada, a aplicação utiliza os valores default configurados na conta.

Ref. 03.01: “tempoExpiracaoDocumentoDias” - é o tempo de expiração em dias para que o documento seja assinado pelos signatários. Após este tempo, caso o documento não seja assinado, ele expira e fica indisponível para assinatura.

Ref. 03.02: “avisoAntesExpiracaoDocumentoDias” - é o tempo em dias que a aplicação usará como referência para começar a notificar o usuário antes da expiração do documento.

Ref. 03.03: “frequenciaLembreteDias” - é a frequência para envio de lembrete aos signatários para lembrá-los de assinar o documento.

Ref. 04: “idPasta” - é o ID de pasta onde o documento será armazenado dentro da aplicação ArqSIGN.

Ref. 05: “renovacaoMeses” - é o tempo em meses contados a partir da data de assinatura do documento, que o mesmo deve ser renovado, gerando lembretes de renovação para o responsável pelo documento.

Ref. 06: “assinarOrdemDestinatarios” – quando “true”, significa que os signatários deverão assinar o documento obedecendo uma ordem definida no campo “ordem”. Quando “false” significa que os signatários não deverão assinar obedecendo uma ordem.

Ref. 07: “nome” - é o nome do processo de assinatura. Este nome é a referência de busca do documento dentro da aplicação.

Ref. 08: “mensagemPadrao” - é a mensagem a ser enviada na notificação para todos os signatários que receberão o documento para ser assinado via Email e que não possuem configuração de “mensagemPersonalizada”.

Ref. 08.01: “titulo” - neste campo deve ser informado o texto que será enviado como assunto da mensagem.

Ref. 08.02: “texto” - neste campo deve ser informado o texto que será enviado como corpo da mensagem.

Ref. 09: “idResponsavel” - é o ID de um usuário ativo vinculado a conta na Plataforma ArqSIGN que será considerado como responsável pelo documento que está sendo enviado via API. Este usuário receberá notificações de conclusão, recusa de assinatura e lembrete de renovação do documento, caso exista.

Ref. 10: “Destinatarios” - nesta parte do JSON devem ser definidos todos os destinatários para o documento em questão. Os destinatários podem ser pessoas que irão assinar o documento ou que receberão uma cópia ao final do processo após a assinatura de todos os signatários.

Ref. 10.01: “ordem” – neste campo deve ser enviada a ordem que o signatário irá assinar o documento. Este campo deve possuir valor, caso o campo Ref. 06 “assinarOrdemDestinatario” tenha sido enviado como true. Os destinatários configurados para receberem uma cópia (destinatários do processo que não assinarão o documento) terão a ordem desconsiderada.

Ref. 10.02: “nome” – neste campo deve ser enviado o nome do destinatário.

Ref. 10.03: “idTipoAção” – neste campo deve ser definido o que o destinatário deverá executar no documento.

- 1 = Assinar Online.
- 2 = Receber uma cópia.

Ref. 10.04: “idTipoAssinatura” – neste campo deve ser definido o tipo de assinatura do destinatário. Quando o “idTipoAção” = 1 (Assinar online), enviar o código de 1 a 4. Quando o “idTipoAção” = 2 (Receber uma cópia), enviar o código 5.

- 1 = Assinatura Eletrônica.
- 2 = Certificado Digital – ICP Brasil.
- 4 = Certificado Digital – Outros.
- 5 = Sem Assinatura

Ref. 10.05: “idFormaEnvio” – neste campo deve ser enviado como o destinatário deverá receber o documento.

- 1 = Email
- 2 = WhatsApp

Ref. 10.06: “email” - caso o “idFormaEnvio” seja 1 (email), neste campo deve ser enviado o email que o destinatário receberá o documento.

Ref. 10.07: “telefone” - caso o “idFormaEnvio” seja 2 (whatsapp), neste campo deve ser enviado o telefone que o destinatário receberá o documento.

Ref. 10.08: “seguranca” - nesta parte do JSON pode ser enviado um código de segurança para o usuário informar no momento de assinar o documento. Veja abaixo quais informações precisam ser enviadas para que isso ocorra:

Ref. 10.08.01: “idMeioEnvio” - neste campo deve ser informado como o código de segurança deve ser enviado para o destinatário. As opções são:

- 1 = SMS (Somente para telefones do Brasil)
- 2 = WhatsApp
- 3 = Email
- 4 = Não enviar (Esta opção deve ser escolhida, quando quem está enviando o documento ao destinatário passará a informação do código de segurança de outra forma, sem usar os meios de notificações da plataforma ArqSIGN).

Ref. 10.08.02: “codigoSeguranca” - neste campo deve ser informado o código de segurança para o destinatário em questão.

Ref. 10.08.03: “codigoSegurancaEmail” - caso tenha enviado no campo “idMeioEnvio” a opção 3 = Email, deve-se informar o email para onde a plataforma ArqSIGN enviará o código de segurança.

Ref. 10.08.04: “codigoSegurancaTelefone” - caso tenha enviado no campo “idMeioEnvio” a opção 1 = SMS ou 2 = WhatsApp, deve-se informar o telefone para onde a plataforma ArqSIGN enviará o código de segurança.

Ref. 10.08.05: “reenviarCodigo” - caso tenha enviado no campo “idMeioEnvio” a opção 1 = SMS ou 2 = WhatsApp, deve-se informar neste campo se o destinatário ao receber o documento para assinar poderá solicitar o reenvio do código. Este reenvio será computado nos

créditos de SMS e WhatsApp da conta. Para permitir o reenvio do código, envie a informação 1 = True, para não permitir o reenvio, envie a informação 0 = False.

Ref. 10.09: “mensagemPersonalizada” - nesta parte do JSON pode se enviar uma mensagem personalizada para o signatário em questão, caso ele receba o documento para assinar por e-mail.

Ref. 10.09.01: “titulo” - neste campo deve ser informado o texto que será enviado como assunto da mensagem.

Ref. 10.09.02: “texto” - neste campo deve ser informado o texto que será enviado como corpo da mensagem.

Ref. 10.10: “destinatariosEntradaDto” - nesta parte do JSON devem ser enviadas as configurações de dados que serão solicitados ao signatário no momento da assinatura.

Ref. 10.10.01: “infComplNomeSignatarioObrigatorio” - envie 1 = true para obrigar o signatário a informar o nome completo, envie 0 = false para não obrigar o signatário a informar o nome completo.

Ref. 10.10.02: “infComplDocumentoSignatarioObrigatorio” - envie 1 = true para obrigar o signatário a informar um documento de identificação de PF, envie 0 = false para não obrigar o signatário a informar um documento de identificação de PF.

Ref. 10.10.03: “infComplRazaoSocialObrigatorio” - envie 1 = true para obrigar o signatário a informar o nome da empresa, envie 0 = false para não obrigar o signatário a informar o nome da empresa.

Ref. 10.10.04: “infComplIdentificadorObrigatorio” - envie 1 = true para obrigar o signatário a informar um documento de identificação de PJ, envie 0 = false para não obrigar o signatário a informar um documento de identificação de PJ.

Ref. 10.10.05: “idInfComplTipoDocumentoSignatarioPF” - neste campo deve ser informado o tipo de documento a ser informado como PF, as opções são:

- 1 = CPF
- 2 = CNH
- 3 = RG
- 4 = Outro

Ref.10.10.06: “infComplDocumentoNomePF” - quando no JSON, o campo “idInfComplTipoDocumentoSignatarioPF” estiver com o valor 4 = Outro, é necessário enviar neste campo um nome que irá identificar esta opção “Outro” para o signatário no momento da assinatura.

Ref.10.10.07: “idInfComplTipoDadosIdentificadorPF” - quando no JSON, o campo “idInfComplTipoDocumentoSignatarioPF” estiver com o valor 4 = Outro, é necessário enviar neste campo a informação se o documento de PF deve ser do tipo 1 = Texto ou 2 = Numérico.

Ref.10.10.08: “infComplQtdeCaracteresPF” - quando no JSON, o campo “idInfComplTipoDocumentoSignatarioPF” estiver com o valor 4 = Outro, é necessário enviar neste campo a quantidade de caracteres para a aplicação validar o documento de PF.

Ref. 10.10.09: “idInfComplTipoDocumentoSignatarioPJ” - neste campo deve ser informado o tipo de documento a ser informado como PJ, as opções são:

- 4 = Outro
- 5 = CNPJ

Ref.10.10.10: “infComplDocumentoNomePJ” - quando no JSON, o campo “idInfComplTipoDocumentoSignatarioPJ” estiver com o valor 4 = Outro, é

necessário enviar neste campo um nome que irá identificar esta opção “Outro” para o signatário no momento da assinatura.

Ref.10.10.11: “idInfComplTipoDadosIdentificadorPJ” - quando no JSON, o campo “idInfComplTipoDocumentoSignatarioPJ” estiver com o valor 4 = Outro, é necessário enviar neste campo a informação se o documento de PJ deve ser do tipo 1 = Texto ou 2 = Numérico.

Ref.10.10.12: “infComplQtdeCaracteresPJ” - quando no JSON, o campo “idInfComplTipoDocumentoSignatarioPJ” estiver com o valor 4 = Outro, é necessário enviar neste campo a quantidade de caracteres para a aplicação validar o documento de PJ.

Ref. 10.11: “Anexos” - nesta parte do JSON devem ser enviadas as configurações de arquivos a serem solicitados aos signatários no momento da assinatura.

Ref. 10.11.01: “anexoDocumentoNome” – envie neste campo o nome do arquivo que deseja que o signatário faça o upload. Exemplo: Frente da Identidade, Selfie etc.

Ref. 10.11.02: “anexoObrigatorio” - envie 1 = true para obrigar o signatário a fazer o upload deste anexo, ou 0 = false para não obrigar.

Ref. 10.11.03: “anexoExibirTodosDestinatarios” - envie 1 = true para após a conclusão do processo de assinatura, este anexo estar disponível para todos os signatários ou envie 0 = false para este anexo estar disponível somente para você que enviou o documento para ser assinado e para quem fez o upload do anexo.

Ref. 10.12: “definirPosicaoAssinaturaAutomatica” - neste campo deve ser enviada a informação se a plataforma deverá ou não definir de forma automática o posicionamento da assinatura para o signatário.

Nota: No JSON, todos os destinatários com a ação de “Assinar Online” precisam ter o posicionamento da representação visual de forma automática ou manual. Para inserir o posicionamento automático da assinatura, a aplicação irá inserir uma página em branco ao final do arquivo com os locais onde todos os signatários assinarão. Desta forma, ao enviar um documento a ser assinado via API, deve-se deixar a aplicação definir o posicionamento da assinatura para todos os signatários ou para nenhum. Não é possível uma opção híbrida, um signatário se envia o posicionamento da assinatura e outro a aplicação escolhe automaticamente.

Então neste campo teremos as seguintes opções:

- Null: ao enviar este parâmetro como null, significa que a aplicação não irá definir o posicionamento da assinatura para os signatários de forma automática, ou seja, será necessário enviar os dados de posicionamento no JSON (definirPosicaoAssinaturaManual);
- 1 = PF: ao enviar este parâmetro com valor 1, significa que a aplicação ArqSign irá incluir uma página no final do arquivo com a definição automática da posição da assinatura do tipo Pessoa Física (PF).
- 2 = PJ: ao enviar este parâmetro com valor 2, significa que a aplicação ArqSign irá incluir uma página no final do arquivo com a definição automática da posição da assinatura do tipo Pessoa Jurídica (PJ).
- 3 = PF e PJ: ao enviar este parâmetro com valor 3, significa que a aplicação ArqSign irá incluir uma página no final do arquivo com a definição automática da posição para duas assinaturas, uma do tipo Pessoa Física (PF) e uma do tipo Pessoa Jurídica (PJ).

Ref. 10.13: “definirPosicaoAssinaturaManual” - nesta parte do JSON devem ser enviados os dados de posicionamento da assinatura no documento. Esta parte deve ser enviada

obrigatoriamente se no campo “definirPosicaoAssinaturaAutomatica”, tiver sido enviada a informação “Null”.

Ref. 10.13.01: “pessoaFisica” - se este signatário for assinar como pessoa física, envie nesta parte o posicionamento da assinatura em questão.

Ref. 10.13.01.01: “pagina” - envie o número da página onde a assinatura deve ser posicionada.

Ref. 10.13.01.02: “Altura” - envie a altura em milímetros do tamanho da caixa onde a assinatura será aplicada.

Ref. 10.13.01.03: “Largura” - envie a largura em milímetros do tamanho da caixa onde a assinatura será aplicada.

Ref. 10.13.01.04: “PosicaoX” - envie a posição em milímetros da borda esquerda da página até a borda esquerda da caixa da assinatura.

Ref. 10.13.01.05: “PosicaoY” - envie a posição em milímetros da borda superior da página até a borda superior da caixa da assinatura.

Ref. 10.13.02: “pessoaJuridica” - se este signatário for assinar como pessoa jurídica, envie nesta parte o posicionamento da assinatura em questão.

Ref. 10.13.02.01: “pagina” - envie o número da página onde a assinatura deve ser posicionada.

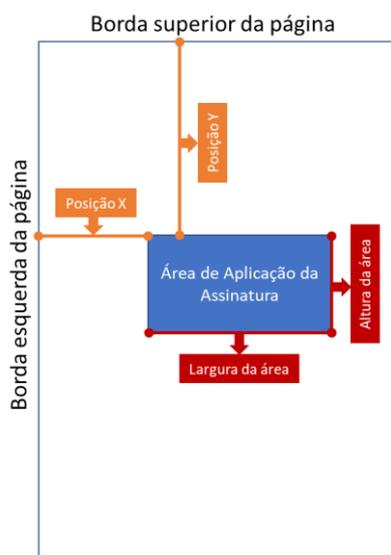
Ref. 10.13.02.02: “Altura” - envie a altura em milímetros do tamanho da caixa onde a assinatura será aplicada.

Ref. 10.13.02.03: “Largura” - envie a largura em milímetros do tamanho da caixa onde a assinatura será aplicada.

Ref. 10.13.02.04: “PosicaoX” - envie a posição em milímetros da borda esquerda da página até a borda esquerda da caixa da assinatura.

Ref. 10.13.02.05: “PosicaoY” - envie a posição em milímetros da borda superior da página até a borda superior da caixa da assinatura.

Nota: Na figura abaixo está a representação visual das medidas que precisam ser identificadas no seu modelo de documento (Altura, Largura, PosicaoX e PosicaoY). Para saber como tirar estas medidas, assista o vídeo pelo link: [Vídeo Explicativo](#).



Ref. 11: “Documento” - nesta parte do JSON deve ser enviado o arquivo a ser assinado. O arquivo deve ser no formato PDF limitado a 100 MB.

Ref. 11.01: “arquivo” - envie neste campo o arquivo em Base64.

Ref. 11.02: “nomeComExtensao” - envie o nome do arquivo com sua extensão.

7.1.2 Retorno

Ref	Response		Body
01	201	Success	{ "IdProcesso": "Guid" }
02	400	Bad Request	“O(s) item(ns) listado(s) é(são) obrigatório(s): “nome dos itens separados por vírgula””.
03			“O(s) item(ns) listado(s) está(ão) com o formato incorreto: “nome dos itens separados por vírgula””.
04			“O(s) id(s) listado(s) não existe(m): “nome dos itens que são Ids de tabela, separados por vírgula””.
05			“Algum parâmetro está incorreto ou é inexistente.”
06			“Saldo de créditos de Envios ou WhatsApp ou SMS insuficiente”.
07			“Arquivo no formato inválido.”
08			“Não é permitido arquivo com tamanho maior que 100MB.”
09			“Conta bloqueada.”
10			401
11	404	Not Found	“Método não encontrado.”
12	500	Server Error	

Detalhamento do Retorno

Ref. 01 – Código 201: Como retorno de sucesso, a aplicação retornará o código 201 e o GUID gerado para o processo. O Guid deve ser armazenado na aplicação do Cliente de forma a possibilitar a usar os métodos de [GET/api/v1/processo/{idprocesso}](#) e [GET/api/v1/processo/{idProcesso}/status-do-processo](#)

Ref. 02 – Código 400: Mensagem de item obrigatório: Esta mensagem será exibida no singular ou plural quando um ou mais itens obrigatórios não tiver sido enviado na chamada da API.

Ref. 03 – Código 400: Mensagem de formato incorreto: Esta mensagem será exibida no singular ou plural quando um ou mais itens estiverem sendo enviados com formato incorreto.

Ref. 04 – Código 400: Mensagem de Ids inexistente: Esta mensagem será exibida no singular ou plural quando um ou mais Id enviado não existir.

Ref. 05 – Código 400: Menagem de parâmetro incorreto ou é inexistente: Quando a chamada é feita com algum parâmetro escrito errado ou parâmetro que não existe no método.

Ref. 06 – Código 400: Mensagem de saldo insuficiente: Esta mensagem será exibida quando a conta não possuir saldo suficiente para o envio por E-mail ou WhatsApp ou SMS.

Ref. 07 – Código 400: Mensagem de arquivo no formato inválido: Esta mensagem será exibida quando o usuário informar um arquivo com extensão diferente de PDF.

Ref. 08 – Código 400: Mensagem de tamanho do arquivo: Esta mensagem será exibida quando o usuário informar um arquivo com tamanho maior que 100MB.

Ref. 09 - Código 400: Mensagem de conta bloqueada: Esta mensagem será exibida quando a conta estiver bloqueada.

Ref. 10 - Código 401: Mensagem de usuário da API não autorizado: AppKey inválida ou não localizada.

7.2 [GET/api/v1/processo/{idprocesso}](#)

7.2.1 Requisição

Header				
Ref.	Parâmetro	Formato	Valor	Requerido
01	Authorization	AppKey	O AppKey identifica a conta.	Sim
Parameters				
Ref.	Parâmetro	Formato	Valor	Requerido
02	idProcesso	"guid"	Id do processo que será consultado.	Sim

Detalhamento do Header e Parameters

Ref. 01: “AppKey” é a chave de autorização para se autenticar na API. Esta chave deve ser válida e estar vinculada a uma conta ArqSIGN ativa.

Ref. 02: “idProcesso” - Para se obter os dados do processo, deve ser enviado como parâmetro o Id do Processo de assinatura na plataforma ArqSIGN. Este ID a API devolve como retorno de sucesso, após a chamada do método: [POST/api/v1/processo/enviar-documento-para-assinar](#)

7.2.2 Retorno

Ref.	Response		Body
01	201	Success	{ JSON com os dados do processo, dados dos signatários, arquivo do processo e registro de assinatura. (Ex. JSON: Tópico 7.2.3) }
02	400	Bad Request	“O(s) item(ns) listado(s) é(são) obrigatório(s): “nome dos itens separados por vírgula””.
03			“O(s) item(ns) listado(s) está(ão) com o formato incorreto: “nome dos itens separados por vírgula””.
04			“O(s) id(s) listado(s) não existe(m): “nome dos itens que são Ids de tabela, separados por vírgula””.
05			“Documento excluído.”
06			“Algum parâmetro está incorreto ou é inexistente.”
07	401	Unauthorized	“Acesso não autorizado.”
08	404	Not Found	“Método não encontrado.”
09	500	Server Error	

Detalhamento do Retorno

Ref. 01 - Código 201: Como retorno de sucesso, a aplicação retornará o código 201 juntamente com os dados completos do processo no formato JSON. Ver exemplo de JSON no tópico 6.2.3.

Ref. 02 - Código 400: Mensagem de item obrigatório: Esta mensagem será exibida no singular ou plural quando um ou mais itens obrigatórios não tiver sido enviado na chamada da API.

Ref. 03 - Código 400: Mensagem de formato incorreto: Esta mensagem será exibida no singular ou plural quando um ou mais itens estiverem sido enviados com formato incorreto.

Ref. 04 - Código 400: Mensagem de Ids inexistente: Esta mensagem será exibida no singular ou plural quando um ou mais Id enviado não existir.

Ref. 05 - Código 400: Mensagem de documento excluído: Esta mensagem será exibida quando o processo em questão tiver sido excluído logicamente.

Ref. 06 - Código 400: Mensagem de parâmetro está incorreto ou é inexistente: Quando a chamada é feita com algum parâmetro escrito errado ou parâmetro que não existe no método.

Ref. 07 - Código 401: Mensagem de usuário da API não autorizado: AppKey inválida ou não localizada.

7.2.3 Exemplo de JSON de Retorno

```
Exemplo Body
{
  "nomeProcesso": "string",
  "idResponsavel": "guid",
  "nomeResponsavel": "string",
  "idPasta": "guid",
  "caminhoDaPasta": "string",
  "status": "string",
  "dataCadastro": "datetime",
  "dataEnvio": "datetime",
  "dataConclusao": "datetime",
  "dataRenovacao": "datetime",
  "dataCancelamento": "datetime",
  "usuarioCancelamento": "string",
  "dataReenvio": "datetime",
  "dataExpiracao": "datetime",
  "expiracaoDias": "smallint",
  "avisoExpiracaoDias": "smallint",
  "lembrete": "true",
  "frequenciaLembrete": "smallint",
  "signatarios":
  [
    {
      "ordem": "tinyint",
      "id": "guid",
      "nome": "string",
      "email": "string",
      "telefone": "string",
      "aguardandoEnvio": "string",
      "falhaEnvio": "string",
      "tipoAcao": "string",
      "copiaEnviada": "string",
      "remetente": "string",
      "assinaturaRecusada": "string",
      "motivoRecusa": "string",
      "seguranca":
      {
        "codigoSeguranca": "string",
        "codigoSegurancaEmail": "string",
        "codigoSegurancaTelefone": "string"
      },
      "dadosAssinatura":
      {
        "tipoAssinatura": "string",
        "dataAssinatura": "datetime",
        "ip": "string",
        "geoLocalizacao": "string",
        "dadosCertificado":
        {
          "nome": "string",
          "emissor": "string",
          "validadeInicio": "string",
          "validadeFim": "string"
        }
      }
    }
  ],
}
```

```

    "dadoPessoaFisica":
    {
      "tipoDocumentoPF": "string",
      "documetoPF": "string",
      "nomePF": "string"
    },
    "dadoPessoaJuridica":
    {
      "tipoDocumentoPJ": "string",
      "documetoPJ": "string",
      "nomePJ": "string"
    },
    "anexos":
    [
      {
        "id": "guid",
        "nome": "string"
      }
    ]
  }
},
"documento":
{
  "nomeComExtensao": "string",
  "contentType": "string",
  "documento": "string base64"
},
"registroAssinaturas":
{
  "nome": "string",
  "arquivo": "string base64"
}
}

```

7.3 GET /api/v1/processo/{idProcesso}/status-do-processo

7.3.1 Requisição

Header				
Ref.	Parâmetro	Formato	Valor	Requerido
01	Authorization	AppKey	O AppKey identifica a conta.	Sim
Parameters				
Ref.	Parâmetro	Formato	Valor	Requerido
02	idProcesso	"guid"	Id do processo que será consultado.	Sim

Detalhamento do Header e Parameters

Ref. 01: “AppKey” é a chave de autorização para se autenticar na API. Esta chave deve ser válida e estar vinculada a uma conta ArqSIGN ativa.

Ref. 02: “idProcesso” - Para se obter o status do processo, deve ser enviado como parâmetro o Id do Processo de assinatura na plataforma ArqSIGN. Este ID a API devolve como retorno de sucesso, após a chamada do método: **POST /api/v1/processo/enviar-documento-para-assinar**

7.3.2 Retorno

Ref	Response		Body
01	201	Success	<pre> { "NomeProcesso": "string", "Status": "string" } </pre>
02	400	Bad Request	“O(s) item(ns) listado(s) é(são) obrigatório(s): “nome dos itens separados por vírgula””.

03			"O(s) item(ns) listado(s) está(ão) com o formato incorreto: "nome dos itens separados por vírgula"".
04			"O(s) id(s) listado(s) não existe(m): "nome dos itens que são lds de tabela, separados por vírgula"".
05			"Documento excluído."
06			"Algum parâmetro está incorreto ou é inexistente."
07	401	Unauthorized	"Acesso não autorizado."
08	404	Not Found	"Método não encontrado."
09	500	Server Error	

Detalhamento do Retorno

Ref. 01 - Código 201: Como retorno de sucesso, a aplicação retornará o código 201 juntamente com o nome e o status do processo.

Ref. 02 - Código 400: Mensagem de Item obrigatório: Esta mensagem será exibida no singular ou plural quando um ou mais itens obrigatórios não tiver sido enviado na chamada da API.

Ref. 03 - Código 400: Mensagem de formato incorreto: Esta mensagem será exibida no singular ou plural quando um ou mais itens estiverem sendo enviados com formato incorreto.

Ref. 04 - Código 400: Mensagem de lds inexistente: Esta mensagem será exibida no singular ou plural quando um ou mais ld enviado não existir.

Ref. 05 - Código 400: Mensagem de Documento excluído: Esta mensagem será exibida quando o processo em questão tiver sido excluído logicamente.

Ref. 06 - Código 400: Mensagem de parâmetro incorreto ou inexistente: Quando a chamada é feita com algum parâmetro escrito errado ou parâmetro que não existe no método.

Ref. 07 - Código 401: Mensagem de usuário da API não autorizado: AppKey inválida ou não localizada.

7.4 GET /api/v1/processo/{idprocesso}/dados-signatarios

7.4.1 Requisição

Header				
Ref.	Parâmetro	Formato	Valor	Requerido
01	Authorization	AppKey	O AppKey identifica a conta.	Sim
Parameters				
Ref.	Parâmetro	Formato	Valor	Requerido
02	idProcesso	"guid"	Id do processo que será consultado os dados dos signatários.	Sim

Detalhamento do Header e Parameters

Ref. 01: "AppKey" é a chave de autorização para se autenticar na API. Esta chave deve ser válida e estar vinculada a uma conta ArqSIGN ativa.

Ref. 02: "idProcesso" - Para se obter o status do processo, deve ser enviado como parâmetro o Id do Processo de assinatura na plataforma ArqSIGN. Este ID a API devolve como retorno de sucesso, após

a chamada do método: **POST**/api/v1/processo/enviar-documento-para-assinar.

Outra forma de obter o id do processo e através da plataforma ArqSIGN disponibilizado na opção “Histórico” do documento.

7.4.2 Retorno

Ref.	Response		Body
01	200	Success	{ JSON com os dados do processo e dos signatários. (Ex. JSON: Tópico 7.4.3) }
02	400	Bad Request	“O(s) item(ns) listado(s) é(são) obrigatório(s): “nome dos itens separados por vírgula””.
03			“O(s) item(ns) listado(s) está(ão) com o formato incorreto: “nome dos itens separados por vírgula””.
04			“O(s) id(s) listado(s) não existe(m): “nome dos itens que são ids de tabela, separados por vírgula””.
05			“Documento excluído.”
06			“Algum parâmetro está incorreto ou é inexistente.”
07	401	Unauthorized	“Acesso não autorizado.”
08	404	Not Found	“Método não encontrado.”
09	500	Server Error	

Detalhamento do Retorno

Ref. 01 - Código 200: Como retorno de sucesso, a aplicação retornará o código 200 juntamente com os dados dos signatários do processo no formato JSON. Ver exemplo de JSON no tópico 7.4.3.

Ref. 02 - Código 400: Mensagem de item obrigatório: Esta mensagem será exibida no singular ou plural quando um ou mais itens obrigatórios não tiver sido enviado na chamada da API.

Ref. 03 - Código 400: Mensagem de formato incorreto: Esta mensagem será exibida no singular ou plural quando um ou mais itens estiverem sido enviados com formato incorreto.

Ref. 04 - Código 400: Mensagem de Ids inexistente: Esta mensagem será exibida no singular ou plural quando um ou mais Id enviado não existir.

Ref. 05 - Código 400: Mensagem de documento excluído: Esta mensagem será exibida quando o processo em questão tiver sido excluído logicamente.

Ref. 06 - Código 400: Mensagem de parâmetro está incorreto ou é inexistente: Quando a chamada é feita com algum parâmetro escrito errado ou parâmetro que não existe no método.

Ref. 07 - Código 401: Mensagem de usuário da API não autorizado: AppKey inválida ou não localizada.

7.4.3 Exemplo de JSON de Retorno

Exemplo Body
<pre>{ "nomeProcesso": "string", "status": "string", "dataExpiracao": "datetime", "signatarios": [{ "ordem": "tinyint", "IdProcessoDestinatario": "guid", "nome": "string", "idTipoAssinatura": "tinyint", }] }</pre>

```

    "IdFormaEnvioProcesso": "bit",
    "email": "string",
    "telefone": "string",
    "tipoAcao": "string",
    "IdMeioEnvioCodigoSeguranca": "bit",
    "emailSeguranca": "string",
    "telefoneSeguranca": "string",
    "permitirReenviarCodigo": "bit",
    "falhaEnvio": "string",
    "assinado": "bit"
  }
}

```

7.5 PATCH /api/v1/processo/{idProcesso}/reenviar-processo

7.5.1 Requisição

Header				
Ref.	Parâmetro	Formato	Valor	Requerido
01	Authorization	AppKey	O AppKey identifica a conta.	Sim
Parameters				
Ref.	Parâmetro	Formato	Valor	Requerido
02	idProcesso	"guid"	Id do processo que será reenviado.	Sim
Body				
Ref	Parametro	Formato	Valor	Requerido
03	"Destinatários"			
03.01	"idProcessoDestinatario"	"Guid"	Id do destinatário que será reenviado o documento	Sim* ³ Se houver dados
03.02	"idFormaEnvioProcesso"	"bit"	Forma de envio do processo 1 - E-mail 2 - Whatsapp	Não
03.03	"nome"	"string"	Nome do destinatário	Não
03.04	"email"	"string"	Email para reenviar o documento. * ¹ Obrigatório se idFormaEnvio = 1	Não* ¹
03.05	"telefone"	"string"	Telefone para reenviar o documento. * ² Obrigatório se idFormaEnvio = 2	Não* ²
03.06	"idMeioEnvioCodigoSeguranca"	"bit"	1 - SMS (Somente Brasil) 2 - Whatsapp 3 - Email 4 - Não enviar	Não
03.07	"emailSeguranca"	"string"	Email para enviar o código de segurança. * ³ Obrigatório se idMeioEnvio = 3.	Não* ³
03.08	"telefoneSeguranca"	"string"	Telefone para enviar o código de segurança. * ⁴ Obrigatório se idMeioEnvio = 1 ou 2	Não* ⁴
03.09	"permitirReenviarCodigo"	"bit"	1 = true 0 = False	Não

Geral: Os parâmetros não obrigatórios listados abaixo devem constar no JSON, e caso não seja necessário para o documento, enviar com o valor null na frente:

- ➔ Ref. 03: "Destinatarios" - Caso não informe os destinatários, o serviço reenvia o processo para todos os destinatários participantes do processo com ação de assinar online e que estejam pendentes de assinaturas na ordem atual.

→ Ref. 03.02 “idFormaEnvioProcesso” - é possível alterar a forma de entrega do documento, mas não é obrigatório

→ Ref. 03.03 “nome”

→ Ref. 03.06 “idMeioEnvioCodigoSeguranca” - é possível alterar a forma de entrega tanto do código de segurança, mas não é obrigatório.

Atenção: Não é permitido inserir código de segurança para destinatários que não possui configuração de código de segurança. Apenas alterar a forma de entrega do código de segurança para os destinatários que possuem a configuração de código de segurança.

→ Ref. 03.09 “permitirReenviarCodigo”

Detalhamento do Header, Parameters e Body

Ref. 01: “AppKey” é a chave de autorização para se autenticar na API. Esta chave deve ser válida e estar vinculada a uma conta ArqSIGN ativa.

Ref. 02: “idProcesso” - Para reenviar o processo, deve ser enviado como parâmetro o Id do Processo de assinatura na plataforma ArqSIGN. Este ID a API devolve como retorno de sucesso, após a chamada do método: [POST/api/v1/processo/enviar-documento-para-assinar](#).

Outra forma de obter o id do processo e através da plataforma ArqSIGN disponibilizado na opção “Histórico” do documento.

Ref. 03: “Destinatarios” - nesta parte do JSON devem ser definidos os destinatários, os quais serão reenviado o documento, podendo ser alterados ou não.

Ref. 03.01: “idProcessoDestinatario” - neste campo deve ser enviado o Id do destinatário que será reenviado o documento.

Ref. 03.04: “email” - quando no JSON, o campo “idFormaEnvio” estiver com o valor 1 = E-mail, é necessário enviar neste campo a informação do email que o destinatário recebeu o documento ou o novo email será reenviado o documento .

Ref. 03.05: “telefone” - quando no JSON, o campo “idFormaEnvio” estiver com o valor 2 = WhatsApp, é necessário enviar neste campo a informação do telefone que o destinatário recebeu o documento ou o novo telefone será reenviado o documento .

Ref. 03.07: “emailSeguranca” - quando no JSON, o campo “idMeioEnvioCodigoSeguranca” estiver com o valor 3 = Email, é necessário enviar neste campo a informação do e-mail que o destinatário recebeu o documento ou o novo e-mail será reenviado o documento .

Ref. 03.08: “telefoneSeguranca” quando no JSON, o campo “idMeioEnvioCodigoSeguranca” estiver com o valor 1 = SMS ou 2 = WhatsApp, é necessário enviar neste campo a informação do telefone que o destinatário recebeu o documento ou o novo telefone será reenviado o documento .

7.5.2 Retorno

Ref.	Response		Body
01	200	Success	Documento reenviado com sucesso.
02	400	Bad Request	"O(s) item(ns) listado(s) é(são) obrigatório(s): "nome dos itens separados por vírgula"".
03			"O(s) item(ns) listado(s) está(ão) com o formato incorreto: "nome dos itens separados por vírgula"".
04			"O(s) id(s) listado(s) não existe(m): "nome dos itens que são lds de tabela, separados por vírgula"".
05			"Algum parâmetro está incorreto ou é inexistente."
06	401	Unauthorized	"Acesso não autorizado."
07	404	Not Found	
08	500	Server Error	

Detalhamento do Retorno

Ref. 01 - Código 200: Como retorno de sucesso, a aplicação retornará o código 200 juntamente com a mensagem de documento reenviado com sucesso.

Ref. 02 - Código 400: Mensagem de item obrigatório: Esta mensagem será exibida no singular ou plural quando um ou mais itens obrigatórios não tiver sido enviado na chamada da API.

Ref. 03 - Código 400: Mensagem de formato incorreto: Esta mensagem será exibida no singular ou plural quando um ou mais itens estiverem sendo enviados com formato incorreto.

Ref. 04 - Código 400: Mensagem de lds inexistente: Esta mensagem será exibida no singular ou plural quando um ou mais ld enviado não existir.

Ref. 05 - Código 400: Mensagem de parâmetro está incorreto ou é inexistente: Quando a chamada é feita com algum parâmetro escrito errado ou parâmetro que não existe no método.

Ref. 07 - Código 401: Mensagem de usuário da API não autorizado: AppKey inválida ou não localizada.

7.6 PATCH /api/v1/processo/{idProcesso}/cancelar-processo

7.6.1 Requisição

Header				
Ref.	Parâmetro	Formato	Valor	Requerido
01	Authorization	AppKey	O AppKey identifica a conta.	Sim
Parameters				
Ref.	Parâmetro	Formato	Valor	Requerido
02	idProcesso	"guid"	Id do processo que será cancelado.	Sim

7.6.2 Retorno

Ref.	Response		Body
01	200	Success	"Documento cancelado com sucesso"
02	400	Bad Request	"O(s) item(ns) listado(s) é(são) obrigatório(s): "nome dos itens separados por vírgula"."
03			"O(s) item(ns) listado(s) está(ão) com o formato incorreto: "nome dos itens separados por vírgula"."
04			"O(s) id(s) listado(s) não existe(m): "nome dos itens que são lds de tabela, separados por vírgula"."
05			"Documento excluído."
06			"Algum parâmetro está incorreto ou é inexistente."
7	422	Unprocessable	Validações tratadas: Documento concluído. Documento cancelado. Documento com status "Criado".
08	401	Unauthorized	"Acesso não autorizado."
09	404	Not Found	"Método não encontrado."
10	500	Server Error	

Detalhamento do Retorno

Ref. 01 - Código 200: Como retorno de sucesso, a aplicação retornará o código 200 juntamente com a mensagem de documento cancelado com sucesso.

Ref. 02 - Código 400: Mensagem de item obrigatório: Esta mensagem será exibida no singular ou plural quando um ou mais itens obrigatórios não tiver sido enviado na chamada da API.

Ref. 03 - Código 400: Mensagem de formato incorreto: Esta mensagem será exibida no singular ou plural quando um ou mais itens estiverem sendo enviados com formato incorreto.

Ref. 04 - Código 400: Mensagem de lds inexistente: Esta mensagem será exibida no singular ou plural quando um ou mais ld enviado não existir.

Ref. 05 - Código 400: Mensagem de documento excluído: Esta mensagem será exibida quando o processo em questão tiver sido excluído logicamente.

Ref. 06 - Código 400: Mensagem de parâmetro está incorreto ou é inexistente: Quando a chamada é feita com algum parâmetro escrito errado ou parâmetro que não existe no método.

Ref. 07 - Código 422: Mensagem validações: Estas mensagens serão exibidas quando o usuário informar um com status diferente de Aguardando ou Em processo.

Ref. 08 - Código 401: Mensagem de usuário da API não autorizado: AppKey inválida ou não localizada.