

Signant.no Web service Instant Signing Functional specification

25. Aug. 2016

Version 1.1.0

Contents

- Version log..... 3
- 1. About Signant.no..... 3
- 2. Introduction..... 3
 - 2.1 Purpose of this document. 3
 - 2.2 Scope of the document 3
 - 2.3 Definitions, Acronyms and Abbreviations..... 3
- 3. DataFlow..... 4
 - 3.1 Dataflow steps..... 4
- 4. Time to Live 4
- 5. Web service functions 5
- Functions description 5
 - 5.1 Function: CreateInstantPosting..... 5
 - Object: InstantPosting..... 5
 - Object: InstantPostingRecipient..... 6
 - Object: Attachment..... 6
 - CreateInstantPosting output:..... 7
 - Object: CreateInstantPostingResponse..... 7
 - Object: UrlToSigning..... 7
 - 5.2 Function GetPostingStatus 7
 - GetPostingStatus output: 8
 - Object: GetPostingStatusResponse..... 8
 - 5.3 Function: DownloadInstantAttachment..... 8
 - DownloadInstantAttachment output..... 9
 - Object: DownloadInstantAttachmentResponse..... 9
- 6. Sample requests 9
 - 6.1. Sample request: CreateInstantPosting..... 9
 - 6.2. Sample request: GetPostingStatus..... 10
 - 6.3. Sample request: DownloadInstantAttachment..... 10

Version log

Version	Date	Changes
1.0	2014.10.03	Initial document
1.0.1	2015.03.03	Small changes
1.1.0	2016.08.25	New functionality: Age and date of birth control added to DateOfBirthRequired object. New field: CancelUrl

1. About Signant.no

Signant.no is a website created and owned by Maestro Soft AS, it allows persons and companies to sign and exchange documents in a simple and efficient way.

With this web service you will get access to functions to create a request for signature, check the request status and download signed documents.

2. Introduction

2.1 Purpose of this document.

Outline the technical solution for initializing and handling a request for signature using the signant.no WS API.

2.2 Scope of the document

This document is meant to describe available functions and dataflow between customer and signant.no system.

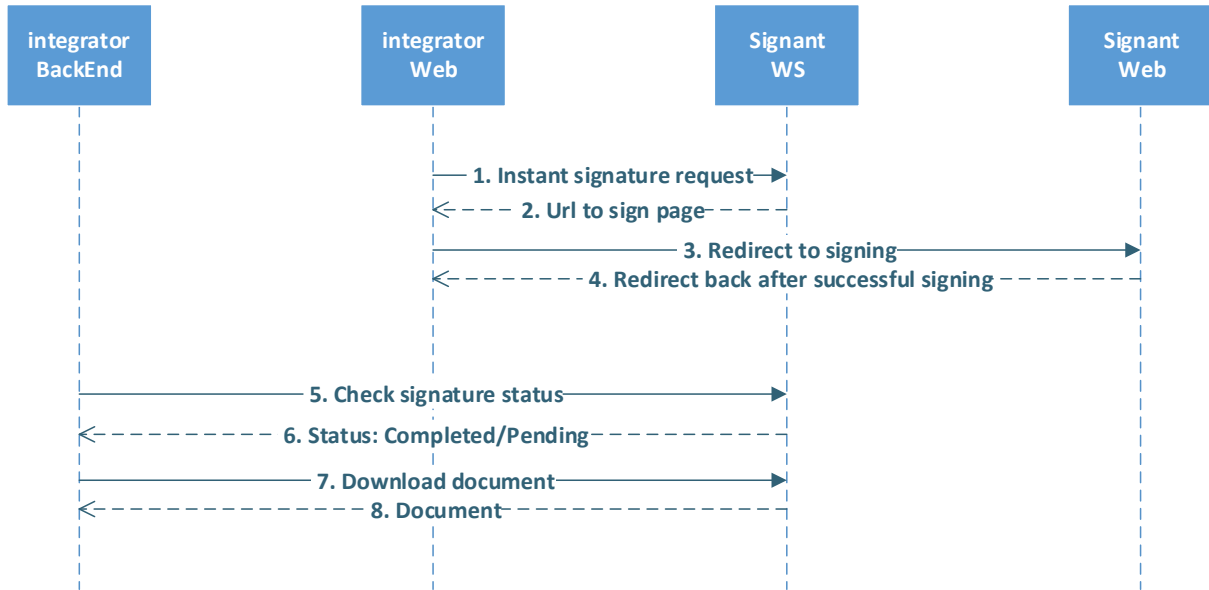
2.3 Definitions, Acronyms and Abbreviations

The following table presents the most significant terms used in this document:

Term	Definition
Signant	Service provider
Integrator	Company using the WS
User	Person that signs the document
Recipient	Recipient of signature request
SSN	Social security number.
PDF	Portable Document Format, read more at: www.adobe.com/pdf
Signature widget	A visible icon in the PDF document indicating each signature.

3. DataFlow

The image below displays a possible scenario in which the “Instant Sign” WS can be used.



3.1 Dataflow steps

- 1) Integrator web initializes a new “Instant Sign” posting.
- 2) Url to signing is returned.
- 3) User is redirected to signature page.
- 4) After successful signing the user is redirected back to Integrator web.
- 5) Integrator backend checks status (every 5 minutes), or when user is redirected back to Integrator web.
- 6) Signant WS returns signature status.
- 7) When signature status is completed Integrator backend downloads the document.
- 8) Signant WS returns the signed document.

4. Time to Live

“Instant Sign” service is meant to be used when a request for signature is required instantly, typically on a web page when the customer is accepting terms of an agreements, signing a contract or similar. The posting and all files related to the posting will therefore be removed from Signant servers after a short period of time.

How posting expiry date is set:

- Upon creation the posting will expire after 15 minutes.
- When user begins the signing process the posting expiry will be set to 15 minutes from that time.

- When signing process is done the posting expiry will again be set to 15 from that time.

Example of a posting expired date being moved 15 minutes for each time an action is executed.



It is crucial for the “Integrator backend” to check the status regularly (every 5 min) and download the document as soon as the WS returns a status indicating that the document has been signed. If this step is not executed correctly the Integrator can potentially lose the signed documents because it will be deleted from Signant servers when the expired date has been reached.

5. Web service functions

This web service exposes the following functions:

Name	Description
CreateInstantPosting	For creating a new instant sign posting.
GetPostingStatus	Returns a simple status overview for the posting.
DownloadInstantAttachment	Download attachment for archiving

Functions description

5.1 Function: CreateInstantPosting

Used to create a new posting. See [6.1](#) for sample request.

Parameters:

Name	Type	Description	Required
DistributorID	String	For authentication (Provided by Signant)	Yes
AccessCode	String	For authentication (Provided by Signant)	Yes
Posting	InstantPosting	InstantPosting object	Yes

Object: InstantPosting

Fields:

Name	Type	Description	Required
DistributorSystemID	String	External system reference, for tracking what system is accessing the service, for example: WebDisclaimer or OrderTerms	No
Title	String	Title of the posting.	Yes
Description	String	Short description of posting	No
UseWidget	Bool	Add visible signature widget in PDF document	Yes
Recipients	InstantPostingRecipient()	List of InstantPostingRecipient	Yes
AttachmentToSign	Attachment	Attachment object	
AfterSignAction	Enum	Specifies what action should be performed after successful signing. Valid values are: NoAction, Redirect, CloseWindow 1)	Yes
AfterSignRedirectUrl	String	Redirect URL if AfterSignAction is set to Redirect	No
AfterSignRedirectDelay	Int	Redirect delay in sec, should not be set if AfterSignRedirectCaption is used	No
AfterSignRedirectCaption	String	If Action is set to Redirect and AfterSignRedirectDelay is not set then this will be the caption on the redirect link	No
CancelUrl	String	Redirect url if the user cancels in the signing process. This must be a https address	No

1) CloseWindow only works if the browser window is opened from a parent window.

Object: InstantPostingRecipient

Fields:

Name	Type	Description	Required
Name	String	Full name of recipient	Yes
Email	String	Email to recipient	No
SSN	String	Social security number of recipient 2)	No
MinimumAge	Int	If this field is set, t will be considered as a constraint for minimum age. Signing process will be cancelled if person does not meet this criteria 3)	No
MaximumAge	Int	If this field is set, it will be considered as a constraint for maximum age. Signing process will be cancelled if person does not meet this criteria 3)	No
DateOfBirthRequired	DateTime	Signing process will be cancelled if person does not meet this criteria 3)	No

2) User of WS must have permission from recipient to use their SSN

3) If this check fails, the user will be redirected to CancelUrl from InstantPosting object (if that parameter is set).

Object: Attachment

Fields:

Name	Type	Description	Required
File	Binary	File to sign 4)	Yes
FileName	String	Name of file	Yes
Description	String	Short description of file	No
ActionType	Enum	Must be: Sign	Yes
CancelUrl	String	Redirect if user cancelled the process. 5)	No

4) File for signing must be a valid PDF file

5) BankID requires a https adress.

CreateInstantPosting output:

Object: CreateInstantPostingResponse

Fields:

Name	Type	Description
Success	Bool	Method call Result
Message	String	Error/Validation Message when Success is false
ErrorCode	Int	Error/Validation code when Success is false
PostingID	Guid	ID of newly created posting
AttachmentID	Guid	AttachmentID of attachment created by Signant
SignUrls	UrlToSigning ()	Array of UrlToSigning , one for each recipient

Object: UrlToSigning

Fields:

Name	Type	Description
Name	String	Name of person to sign (from InstantPostingRecipient)
Url	String	Url to signature page

5.2 Function GetPostingStatus

Used to check posting status. When function returns "Incorrect postingID/ErrorCode: 4008" posting is expired and has been deleted from the system. See [6.2](#) for sample request.

Parameters:

Name	Type	Description	Required
DistributorID	String	Provided by Signant	Yes
AccessCode	String	Provided by Signant	Yes
PostingID	Guid	Posting ID	Yes

GetPostingStatus output:

Object: GetPostingStatusResponse

Fields:

Name	Type	Description
Success	Bool	Method call Result
Message	String	Error/Validation Message when Success is false
ErrorCode	Int	Error/Validation code when Success is false
PostingID	Guid	Posting ID
Status	Enum	Sent, Completed, CompletedPartially, Expired
TotalDocsToSign	Int	Total documents to be signed
TotalDocsSigned	Int	Document count where recipients have signed
TotalDocsRejected	Int	Document count where at least one recipient has rejected
ToBeSignedCount	Int	Document signing request count.
SignatureCount	Int	Number of completed signatures.
Activated	Bool	Posting Activation Status
PostingType	Enum	For Instant Sign this will be: InstantSign
ActiveTo	DateTime	Posting Expiration DateTime
CreatedDateTime	DateTime	Posting Creation DateTime
LastModifiedDateTime	DateTime	Posting Modified DateTime
WillBeDeletedDateTime	DateTime	Date when the posting will to be deleted
Title	String	Title of the posting
Description	String	Short description on
DefaultLanguage	String	Default Language Code
CompletedDateTime	DateTime	Posting completed date time
LastReminderDateTime	DateTime	Not in use
TotalDocsToRead	Int	Not in use
TotalDocsRead	Int	Not in use
TotalDocsToUpload	Int	Not in use
TotalDocsUploaded	Int	Not in use

5.3 Function: DownloadInstantAttachment

For downloading the attachment after successful signing. Attachment cannot be downloaded after posting has expired. See [6.3](#) for sample request.

Parameters:

Name	Type	Description	Required
DistributorID	String	Provided by Signant	Yes
AccessCode	String	Provided by Signant	Yes
PostingID	Guid	Posting ID	Yes
AttachmentID	Guid	Attachment ID	Yes

DownloadInstantAttachment output

Object: DownloadInstantAttachmentResponse

Fields:

Name	Type	Description
Success	Bool	Method call Result
Message	String	Error/Validation Message when Success is false
ErrorCode	Int	Error/Validation code when Success is false
AttachmentID	Guid	Attachment ID
FileName	String	Name for file
AttachmentFile	Byte[]	The file
Description	String	Attachment Description
FileHash	String	Md5 checksum of the file

6. Sample requests

The sample requests and responses have been shortened down for readability and will therefore not work “as is”, you will have to replace all required fields with correct data before trying to send a request to the service.

6.1. Sample request: CreateInstantPosting

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <CreateInstantPosting xmlns="http://tempuri.org/">
      <DistributorID>Your DistributorID</DistributorID>
      <AccessCode>Your AccessCode</AccessCode>
      <Posting xmlns:a="http://schemas.datacontract.org/2004/07/Signant.Common.Models"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:AfterSignAction>Redirect</a:AfterSignAction>
        <a:AfterSignRedirectCaption i:nil="true"/>
        <a:AfterSignRedirectDelay>0</a:AfterSignRedirectDelay>
        <a:AfterSignRedirectUrl>https://www.google.com/</a:AfterSignRedirectUrl>
        <a:CancelUrl>https://www.google.com/cancel</a:CancelUrl>
        <a:AttachmentToSign>
          <a:ActionType>Sign</a:ActionType>
          <a:Description>Description of the file</a:Description>
          <a:File>..Bytes..</a:File>
          <a:FileName>Contract2014.pdf</a:FileName>
        </a:AttachmentToSign>
        <a:Description i:nil="true"/>
        <a:DistributorSystemID i:nil="true"/>
        <a:Recipients>
          <a:InstantPostingRecipient>
            <a:Email i:nil="true"/>
            <a:Name>Kari Nordmann</a:Name>
            <a:SSN i:nil="true"/>
            <a:MinimumAge i:nil="true"/>
            <a:MaximumAge i:nil="true"/>
            <a:DateOfBirthRequired i:nil="true"/>
          </a:InstantPostingRecipient>
        </a:Recipients>
        <a>Title>This is the title of the posting</a>Title>
        <a:UseWidget>true</a:UseWidget>
      </Posting>
    </CreateInstantPosting>
  </s:Body>
</s:Envelope>
```

The request above will return the following result:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <CreateInstantPostingResponse xmlns="http://tempuri.org/">
      <CreateInstantPostingResult xmlns:a="http://schemas.datacontract.org/2004/07/Signant.Common.Models"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:AttachmentID>ea94460e-bd4f-42be-89c5-627ce205366b</a:AttachmentID>
        <a:ErrorCode>0</a:ErrorCode>
        <a:Message>OK</a:Message>
        <a:PostingID>3b57348e-605e-4a66-acf4-8789828ef9ee</a:PostingID>
        <a:SignUrls>
          <a:UrlToSigning>
            <a:Name>Kari Nordmann</a:Name>
            <a:Url>https://someurl.com/InstantSign</a:Url>
          </a:UrlToSigning>
        </a:SignUrls>
        <a:Success>>true</a:Success>
      </CreateInstantPostingResult>
    </CreateInstantPostingResponse>
  </s:Body>
</s:Envelope>
```

6.2. Sample request: GetPostingStatus

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetPostingStatus xmlns="http://tempuri.org/">
      <DistributorID>Your DistributorID</DistributorID>
      <AccessCode>Your AccessCode</AccessCode>
      <PostingID>329218e9-0da8-42fc-98eb-98e4ee02bc5d</PostingID>
    </GetPostingStatus>
  </s:Body>
</s:Envelope>
```

The request above will return the following result:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetPostingStatusResponse xmlns="http://tempuri.org/">
      <GetPostingStatusResult xmlns:a="http://schemas.datacontract.org/2004/07/Signant.Common.Models"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:Activated>true</a:Activated>
        <a:ActiveToDateTime>2014-10-02T12:16:07.303</a:ActiveToDateTime>
        <a:CompletedDateTime>0001-01-01T00:00:00</a:CompletedDateTime>
        <a:CreatedDateTime>2014-10-03T12:01:07.307</a:CreatedDateTime>
        <a:DefaultLanguage>NO</a:DefaultLanguage>
        <a:Description/>
        <a:ErrorCode>0</a:ErrorCode>
        <a:LastModifiedDateTime>2014-10-02T12:01:07.307</a:LastModifiedDateTime>
        <a>LastReminderDateTime>0001-01-01T00:00:00</a>LastReminderDateTime>
        <a:Message>OK</a:Message>
        <a:PostingID>329218e9-0da8-42fc-98eb-98e4ee02bc5d</a:PostingID>
        <a:PostingType>InstantSign</a:PostingType>
        <a:SignatureCount>0</a:SignatureCount>
        <a>Status>Sent</a>Status>
        <a:Success>true</a:Success>
        <a>Title>This is the title of the posting</a>Title>
        <a:ToBeSignedCount>1</a:ToBeSignedCount>
        <a>TotalDocsRead>1</a>TotalDocsRead>
        <a>TotalDocsRejected>0</a>TotalDocsRejected>
        <a>TotalDocsSigned>0</a>TotalDocsSigned>
        <a>TotalDocsToRead>0</a>TotalDocsToRead>
        <a>TotalDocsToSign>1</a>TotalDocsToSign>
        <a>TotalDocsToUpload>0</a>TotalDocsToUpload>
        <a>TotalDocsUploaded>0</a>TotalDocsUploaded>
        <a:WillBeDeletedDateTime>2014-10-02T12:16:07.303</a:WillBeDeletedDateTime>
      </GetPostingStatusResult>
    </GetPostingStatusResponse>
  </s:Body>
</s:Envelope>
```

6.3. Sample request: DownloadInstantAttachment

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
```

```
<DownloadInstantAttachment xmlns="http://tempuri.org/">
  <DistributorID>Your DistributorID</DistributorID>
  <AccessCode>Your DistributorID</AccessCode>
  <PostingID>329218e9-0da8-42fc-98eb-98e4ee02bc5d</PostingID>
  <AttachmentID>92a1df59-a09c-4621-a534-10049d101f68</AttachmentID>
</DownloadInstantAttachment>
</s:Body>
</s:Envelope>
```

The request above will return the following result:

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <DownloadInstantAttachmentResponse xmlns="http://tempuri.org/">
      <DownloadInstantAttachmentResult xmlns:a="http://schemas.datacontract.org/2004/07/Signant.Common.Models"
        xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <a:AttachmentFile>..Bytes..</a:AttachmentFile>
        <a:AttachmentID>92a1df59-a09c-4621-a534-10049d101f68</a:AttachmentID>
        <a:Description>Description of the file</a:Description>
        <a:ErrorCode>0</a:ErrorCode>
        <a:FileHash>5c19bbe3b6bfa3d0a0435bfc58d8d9e2</a:FileHash>
        <a:FileName>Contract2014.pdf</a:FileName>
        <a:Message>OK</a:Message>
        <a:Success>true</a:Success>
      </DownloadInstantAttachmentResult>
    </DownloadInstantAttachmentResponse>
  </s:Body>
</s:Envelope>
```