

Tutorial: Hello World



Attention

Please note that this documentation and the links refer to the api v6. For more information please see the [migration guide](#) and the [documentation related to v5](#).

This tutorial shows how eSignAnyWhere can be implemented. This guide is intended to get to know the basics of a signature request workflow. After this guide you will be able to upload documents, create envelopes and send envelopes. For more information about the REST API you can also start with the [Postman Tutorial](#).

- [What the tutorial is about](#)
- [Hello World](#)
 - [Version request](#)
 - [Upload document](#)
 - [Prepare envelope \(optional\)](#)
 - [Send envelope](#)
 - [Open the envelope](#)
 - [Find the envelope](#)
 - [Envelope information](#)
 - [Callback](#)
 - [Download finished document](#)

eSignAnyWhere can be easily implemented. This tutorial shows you how to send your first envelope via REST api service of eSignAnyWhere. If you are using Postman for your REST calls please also have a look at the following Tutorial: [Visit postman tutorial](#).

Please also see the developer mode: [Developer mode](#)

For this tutorial you can use your desired programming language (with REST support) or any REST tool (e. g. Postman). Moreover you will need an active eSignAnyWhere account for the authorization. (Even a trial account will work).

Some other interesting documentation links:

- [Developer mode](#)
- [Java Library](#)

For this tutorial please use the following endpoint and the following api documentation link:

REST URI of eSignAnyWhere: <https://demo.esignanywhere.net/Api/v6.0/>

Swagger api documentation: <https://demo.esignanywhere.net/Api>

What the tutorial is about

The tutorial will show you the following basic use-case:



Hello World

Before starting with the first request please note the following:

All api calls need an authorization. You authorize with the

- the apiToken **v 20.42**

The API token is a user specific secret which should not be shared with other users. We recommend to create different API keys for different application integrations, to avoid configuring the same key in various integration systems. This allows, e.g. in case of sharing a key by mistake, to disable one key while keeping other integrations working with their existing configuration.

If you authorize with the apiToken just add the apiToken in the header of the api call. Please see the next sample authorization (Bearer token):

Key	Value
"Authorization"	"Bearer asdfngtmvv8pfmsuaxpzz85zux3e63dd9ztrwix9mln6qka6tds83du3p3lroe"

Please see the next sample authorization (api token):

Key	Value
"ApiToken"	"asdfngtmvv8pfmsuaxpzz85zux3e63dd9ztrwix9mln6qka6tds83du3p3lroe"

Version request

First we will try the connection with a version request. You can find the endpoints in the next table:

Method	REST Endpoint
GET	https://demo.esignanywhere.net/Api/v6/system/version

REST Response
"22.20.0.71"

Upload document

 Authorization is required.

 It is not allowed to upload multiple files at once (when more files are selected).

First we have to upload a document. This will return a documentId, to use the document for creating an envelope. Therefore, we will use the following endpoint:

Method	REST Endpoint
POST	https://demo.esignanywhere.net/Api/v6/file/upload

For this tutorial we just use a simple PDF document. You can download it [here](#) or use your own document.

You need the documentId for creating your first envelope. After uploading the file it is just temporary on the server. After 10 minutes it will be deleted and you are not able to use it again. The moment you are creating an envelope with the file, it gets the time-to-live of the envelope. Attention: The returned file id is just for creating an envelope. Once the envelope is created, the file id becomes invalid, so for downloading a finished file, you have to retrieve the id from the envelope status with the following URI for REST: <https://demo.esignanywhere.net/Api/v6.0/envelope/##envelopeld##> (Replace the placeholder ##envelopeld## with your envelope id.

REST Response
"3252931f-1234-420c-1234-0c7656d8d2ea"

Prepare envelope (optional)

 Authorization is required.

This step is only optional, but required if you do not have a configuration. A configuration is a definition of a signing task for one activity. So it contains information about signature fields on the document, form-fields, etc. With the Developer-Feature of eSignAnyWhere 2.6 it is not required anymore.

A configuration is required for each signing step in your workflow. If you have already your configuration you can skip this step.

To receive an adhoc configuration you can use the following endpoint:

Method	REST Endpoint
POST	https://demo.esignanywhere.net/Api/v6/file/prepare

REST Call

```
{
  "FileIds": [
    "3252931f-1234-420c-1234-0c7656d8d2ea"
  ]
}
```

Additional parameters can be added to this call (optional):

REST Call

```
{
  "FileIds": [
    ...
  ],
  "ClearFieldMarkupString": true,
  "SigStringConfigurations": [
    {
      "StartPattern": "string",
      "EndPattern": "string",
      "ClearSigString": true,
      "SearchEntireWordOnly": true
    }
  ]
}
```

The following response shows an empty document with no unassigned elements and no assigned activities. If there are any form fields already on the document you will get the configuration shown with this call.

REST Response

```
{
  "UnassignedElements": {
    "TextBoxes": [],
    "CheckBoxes": [],
    "ComboBoxes": [],
    "RadioButtons": [],
    "ListBoxes": [],
    "Signatures": [],
    "Attachments": [],
    "LinkConfiguration": {
      "HyperLinks": [],
      "DocumentLinks": []
    }
  },
  "Activities": []
}
```

Send envelope



Authorization is required.

For sending an envelope we are using the following endpoint:

Method	REST Endpoint
--------	---------------

POST

<https://demo.esignanywhere.net/Api/v6/envelope/send>

The envelope configuration defines the envelope and the steps in the workflow. For each action you need to define a configuration, which defines what the signer has to do in his/her signing task.

You can find a sample configuration which contains two actions (one action which contains a signer with ClickToSign and one Action which contains a send copy to the given recipient configuration) for this call in the next section:



The recipient language must be supported by your organization (Settings->Localization).

REST Call

```
{
  "Documents": [{
    "FileId": "d33d43ca-1234-1234-1234-b645fc4e0fb2",
    "DocumentNumber": 1
  }
],
  "Name": "Test",
  "Activities": [{
    "Action": {
      "Sign": {
        "RecipientConfiguration": {
          "ContactInformation": {
            "Email": "jane.doe@sample.com",
            "GivenName": "Jane",
            "Surname": "Doe",
            "LanguageCode": "EN"
          }
        }
      },
      "Elements": {
        "Signatures": [{
          "ElementId": "sample sig click2sign",
          "Required": true,
          "DocumentNumber": 1,
          "DisplayName": "Sign here",
          "AllowedSignatureTypes": {
            "ClickToSign": {
            }
          }
        }
      ],
      "FieldDefinition": {
        "Position": {
          "PageNumber": 1,
          "X": 100,
          "Y": 200
        },
        "Size": {
          "Width": 100,
          "Height": 70
        }
      }
    }
  }
],
  "Action": {
    "SendCopy": {
      "RecipientConfiguration": {
        "ContactInformation": {
          "Email": "john.doe@sample.com",
          "GivenName": "John",
          "Surname": "Doe",
          "LanguageCode": "EN"
        }
      }
    }
  }
}
```

REST Response

```
{
  "EnvelopeId": "45fd01ce-1234-4792-1234-f5230e41b130"
}
```

The envelope id is used for managing the envelopes (send reminder, cancel delete, reject envelope,...).

After the successful creation of the envelope, it is sent to the first recipient.

Open the envelope

After you have sent the envelope and you have got the envelopeld you can call the method <https://demo.esignanywhere.net/api/v6/envelope/{envelopeld}/viewerlinks>. With the result of this method you get information about the envelope and you can find the `workstepRedirectionURL` (Viewerlink).

REST Response

```
{
  "ViewerLinks": [
    {
      "ActivityId": "5609d2ea-1234-1234-1234-1959e63fde64",
      "Email": "john.doe@sample.com",
      "ViewerLink": "https://demo.esignanywhere.net/workstepredirector/sign?
identifier=mdIkzVC1234512IgiOLD3Iodfynd~12345RbJQXusyZuCnp7FR6PniOT12345B~fQ1234A=="
    }
  ]
}
```

With this redirection link it is possible to open and sign the document in your web portal.

Find the envelope



Authorization is required.

After sending the envelope you can use the following api call to search for the envelope:

Method	REST Endpoint
POST	https://demo.esignanywhere.net/Api/v6/envelope/find

In the next section you can find a sample configuration which you need for the api call:

Following status are available:

- Canceled
- Completed
- Expired
- Rejected
- ActionRequired
- WaitingForOthers
- ExpiringSoon
- Active

REST Call

```
{
  "StartDate": "2022-12-05T13:09:53.052Z",
  "EndDate": "2022-12-05T13:09:53.052Z",
  "SearchText": "HelloWorld",
  "Status": "Canceled",
  "InStatusSinceDays": 10,
  "SenderEmail": "john.doe@sample.com",
  "SignerEmail": "jane.doe@sample.com",
  "RecipientEmail": "jane.doe@sample.com",
  "WaitingForRecipientWithEmail": "jane.doe@sample.com",
  "BulkParentId": "83467fdc-1234-4592-1234-9c64787cd7a1"
}
```

Please note: The SearchText searches texts within the following elements:

- Recipient
 - First name
 - Last name
 - Email address
 - Email body
- Envelope
 - Subject
 - Email body
 - Envelope name
 - Envelope description
- Sender
 - First name
 - Last name
 - Email address
 - User id

Please note that the configuration above shows all available filters you may want to use. You can also just search for example for completed envelopes. Therefore only the following configuration is needed:

REST Call

```
{
  "Status": "Completed",
}
```

You can search for the following status:

Status
Canceled
Completed
Expired
Rejected
ActionRequired
WaitingForOthers
ExpiringSoon
Active

If you have not finished the envelope then search for "Active" envelopes, if the envelope is already finished please search for "Completed" envelopes to find the envelope you have sent in the api call above.

As response from this api call you will get a list of all envelopes which fulfill the status. Please see the following sample response:

REST Response

```
{
  "Envelopes": [
    {
      "Status": "Completed",
      "Id": "62664b59-1234-455c-1234-3ed972fe5857",
      "Name": "Test2",
      "BulkParentId": "",
      "IsExpiringSoon": false
    },
    {
      "Status": "Completed",
      "Id": "45fd01ce-1234-4792-1234-f5230e41b130",
      "Name": "Test",
      "BulkParentId": "",
      "IsExpiringSoon": false
    }
  ]
}
```

Envelope information



Authorization is required.

The envelope id can be used any time to receive the envelope status. Therefore, you simply call:

Method	REST Endpoint
GET	https://demo.esignanywhere.net/Api/v6/envelope/##envelopeId##

You can find a sample response for this call in the next section:

REST Response

```
{
  "Id": "45fd01ce-1234-4792-1234-f5230e41b130",
  "EnvelopeStatus": "Completed",
  "Name": "Test",
  "Activities": [
    {
      "Id": "e99769a9-1234-4e8a-1234-b237d41bf720",
      "Status": "Completed",
      "FinishedDate": "2022-05-06T09:00:33.37+00:00",
      "OpenedDate": "2022-05-06T09:00:16.553+00:00",
      "Action": {
        "Sign": {
          "ContactInformation": {
            "Email": "##EMAIL##",
            "GivenName": "##NAME##",
            "Surname": "##NAME##",
            "LanguageCode": "EN"
          }
        }
      }
    },
    {
      "Id": "be0c0dd0-1234-4d6b-1234-7771138f19e8",
      "Status": "Completed",
      "FinishedDate": "2022-05-06T09:00:39.277+00:00",
      "Action": {
        "SendCopy": {
          "ContactInformation": {
            "Email": "##EMAIL##",
            "GivenName": "##NAME##",
            "Surname": "##NAME##",
            "LanguageCode": "EN"
          }
        }
      }
    }
  ]
}
```

Callback

The configured callback URL is used to notify your system if the state of an envelope or workstep is changing. So you can track the status of envelopes without polling the eSignAnyWhere server regularly.

Please note the following referring callbacks:



Only the envelope callback is fired, when the envelope is in a final state. The status update callback is fired by a sub-component and you may require to wait a post-processing time that the envelope reaches its final state. Therefore, please send back the HTTP 200 immediately! If you are not returning the HTTP 200 immediately, eSignAnyWhere tries to recall the URL. Attention: After some attempts the envelope will not be finished!

Due to these notes please send the message immediately after you receive the callback and before other callback-processing starts (e.g. download documents) to avoid a timeout of the callback!

Download finished document



Authorization is required.

First you need the following api call to get the fields of the document (please note that this method is only for completed envelopes):

Method	REST Endpoint
GET	https://demo.esignanywhere.net/Api/v6/envelope/##envelopeId##/files

REST Response

```
{
  "SentDocuments": [
    {
      "FileName": "test.pdf",
      "PageSizes": [
        {
          "Width": 612.0,
          "Height": 792.0
        }
      ]
    }
  ],
  "FinishedDocuments": [
    {
      "FileId": "806b89db-1234-1234-908b-4f2ba4ebd19e",
      "FileName": "test.pdf",
      "Attachments": []
    }
  ],
  "AuditTrail": {
    "FileId": "5e7b8aba-1234-1234-91f8-2c718bf913b1",
    "XmlFileId": "a10a1438-1234-1234-9799-16ae3ce54c37"
  },
  "Disclaimers": []
}
```

To download a finished document you can call the function:

Method	REST Endpoint
GET	https://demo.esignanywhere.net/Api/v6/file/##fileId##

Just insert one of the following file ids:

- Finished document
 - File id
- Audit trail
 - Filed id
 - XML File id



Additional Information

Depending on the signature method used, it might be necessary or recommended to store additional documents beside the signed document, to ensure having the necessary evidence. For advanced electronic signatures, you might have to store also the audit trail document; and for Namirial Disposable Certificate, also the Certificate Request Form by which the signer requested issuance of the QEC. Those documents can be downloaded using the same method, just with the different file IDs.