

ENFOCUS



REVIEW

Enfocus Review API
documentation

Contents

1. About the Enfocus Review API.....	3
2. Getting started.....	4
2.1. Getting access to Workato.....	4
2.2. Generating an API key.....	5
2.3. Trying an endpoint.....	6
3. Review API Overview.....	7
3.1. Terminology.....	7
3.1.1. Cluster.....	7
3.1.2. Repo ID (repoid).....	7
3.1.3. Site Name (siteName).....	8
3.1.4. File ID (fileid).....	8
3.1.5. Presigned URLs.....	8
3.1.6. Authentication of requests.....	8
3.2. Endpoints.....	9
3.2.1. Upload files.....	9
3.2.2. Delete files.....	16
3.2.3. Retrieve Review Public API Info.....	17
3.2.4. Get Annotations.....	17
3.2.5. Create brand - With logo.....	18
3.2.6. Create brand - Without logo.....	19
3.2.7. Delete the given brand.....	19
3.2.8. Set the properties of the brand.....	20
3.2.9. Fetch the details of a specific brand.....	22
3.2.10. Get the presigned URL for brand logo.....	24
3.2.11. Retrieve and list file information - Fetch file details.....	24
3.2.12. Get a list of all the available collections in the given account.....	26
3.2.13. Retrieve and list file information - List all files.....	27
3.2.14. Update the properties of the brand - Both the logo and the brand properties.....	28
3.2.15. Update the properties of the brand - Without modifying the logo.....	29
4. Copyrights.....	30

1. About the Enfocus Review API



The Enfocus Review API is available to customers and OEM partners to integrate Enfocus Review into their own workflow, MIS, or W2P using a REST API.



Note: Access to the Enfocus Review API is a paid feature and must be explicitly enabled for your account. It also requires a [Review subscription](#).

The API is available on Workato, which serves as an integration layer between Review and your custom solution. The Enfocus API Platform on Workato provides:

- Access to documentation for each endpoint
- A built-in interface to test API calls directly
- The option to export all endpoints to tools like Postman, for further exploration

Just like with the Enfocus Review app in Switch, all API usage is linked to your existing Review subscription. Transactions are deducted from your account amount, but API calls will **not** be blocked if you exceed your subscription amount. If you run out of transactions, you can either upgrade your plan or pay for the overuse at the end of your subscription.

With the Enfocus Review API you can:

- Upload one or more files to Enfocus Review and receive a shareable link
- Retrieve the status and metadata of a PDF
- List and search collections and files in your Enfocus Review site
- Delete uploaded files
- Manage brands: add, remove, and list them



Note: Currently, iFraming the Review Viewer on your custom domain is not possible due to security reasons.

2. Getting started

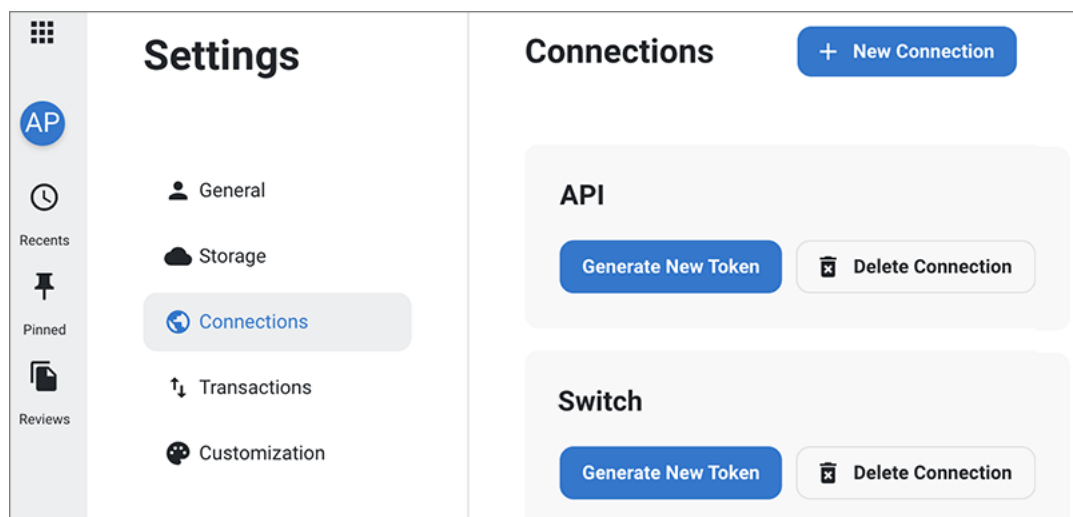
The Enfocus Review API is available through the Workato platform. Learn how you can get access to this platform and how you can create an API key in order to try the API on Workato.

2.1. Getting access to Workato

The Enfocus Review API is available through the Workato platform. Workato acts as an integration layer that simplifies working with the Enfocus API. Instead of managing authentication, error handling, and low-level API calls directly, Workato provides a user-friendly interface for building workflows.

To get access to this platform, proceed as follows:

1. Contact Enfocus to purchase the Enfocus API:
 - If you're a **customer**, contact an [Enfocus reseller](#). You'll receive an email requesting you to generate an Enfocus Cloud Token (see step 2) and provide it to your Enfocus reseller (see step 3).
 - If you're an **OEM partner or prospect**, reach out to your Enfocus OEM partner manager or mail sales@enfocus.com for pricing and next steps.
2. When you receive an email with the request to generate an Enfocus Cloud Token, proceed as follows:
 - a. Open the Enfocus Review Admin Area.
 - b. Generate an Enfocus Cloud Token under **Review Admin > Settings > Connections**.



- To link Workato for API access to your Enfocus Review environment, you must be the Enfocus Review Administrator.
- For the full procedure, refer to [Generating an Enfocus Cloud Token](#).

- We recommend using a meaningful name for this connection, for example "API". In the description, you may add 'Do not delete this connection', as this Enfocus Cloud Token is necessary for your connection to work.



Important: Do not delete this token, as it is required to maintain the connection between your Enfocus Review environment and Workato. Deleting it will require you to re-request API access to Enfocus for the integration with your custom solution.

3. Provide this Cloud Token to Enfocus:

- If you're a **customer**, send the Cloud Token to your Enfocus reseller.
- If you're an **OEM partner or prospect**, send the Cloud Token to the contact person provided by Enfocus.

In return, you'll receive an email from Enfocus, which includes the link to Workato. Allow some time for the email to be sent, since setting up the Workato API access may take some time.

4. Once you receive the email, open it and click the **Get Started** button or the Developer Portal Link.



Tip: Bookmark this URL for later use.

- 5.** In the dialog that appears, enter your email address and click **Send magic link**. Almost instantly you'll receive a mail with a magic link that remains valid for 30 minutes.
- 6.** Click the magic link in the mail.

You now have access to the Enfocus Review API on Workato.



Note: Next time you want to enter the platform, for security reasons, you have to enter your email address again (steps 4-6).

2.2. Generating an API key

In order to try the Enfocus Review API on Workato, you need an API key to connect Workato to your custom integration or solution. This key needs to be entered for each endpoint you want to try.

To generate an API key, proceed as follows:

- 1.** On the Workato platform, do one of the following:
 - Click your account icon in the top right corner and in the menu that appears, click the **Manage API keys** button.
 - Go to **My APIs > Enfocus Review : Endpoints** and click one of the endpoints.
- 2.** Click the **Create API key** button.
- 3.** In the pop-up that is displayed, enter the **API key name**.
Choose something generic, for example: API-Token.
The other fields (Allowed IPs and Blocked IPs) are optional.

4. Click **Generate**.
Your authentication token is generated and displayed.
5. Click the **Copy** button.



Note: Save the authentication token somewhere secure, as it won't be possible to view it once more due to security reasons. However, you can always generate a new one if necessary.

6. To close the dialog, click **Close**.
The API key is stored on the [Manage API keys](#) page. You can refresh it or edit and delete it through the three dots button. Note however that refreshing or deleting the key will have an impact on the connection between Workato and your custom integration or solution.

You are now ready to try the Enfocus Review API. See [Trying an endpoint](#) on page 6.

2.3. Trying an endpoint

Once you have created an API key, you can try the endpoints in Workato.

To do so, proceed as follows:

1. On the details page of an endpoint, make sure the API key is set. Do one of the following:
 - If the **Auth token** field contains a string, you're good to go.
 - If the **Auth token** field is empty, copy the API key (created as described [here](#)) in this field and click **Set token**.
2. Click the **Try it out** button.
3. Enter the appropriate parameters.
Examples are: RepoID and SiteName. For a complete overview, refer to the [API Overview](#).
4. Click **Execute**.
The response is shown below the parameters.

3. Review API Overview

3.1. Terminology

3.1.1. Cluster

The base URLs differ between the US and EU clusters of Review, which you can identify from your Review URL.

- US: <https://portal.us.enfocus.cloud/review>
- EU: <https://portal.eu.enfocus.cloud/review>

In Workato we have the following URLs:

- US: <https://esko-rnd-dev-workato.esko-saas.com>
- EU: <https://esko-apiplatform-eu-dev.esko-saas.com>

3.1.2. Repo ID (repold)

Each account has by default one repository instance. A repository instance has a root node under which all the data for that account can be stored. The repository ID is a base-58 ID that identifies a repository instance.

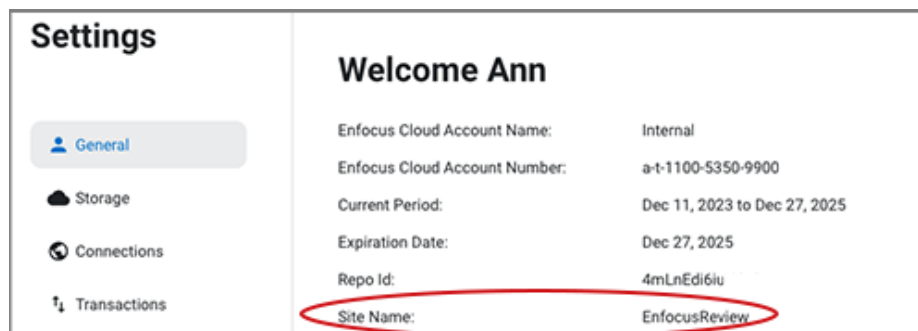
You can find your Repo ID in the General Settings of the Enfocus Admin Area. This is also the ID that you can find in the URL of your Enfocus Review Admin area, for example: <https://portal.eu.enfocus.cloud/review/988nEdi6iuYde6/EnfocusReview/reviews>.

Settings	
Welcome Ann	
Enfocus Cloud Account Name:	Internal
Enfocus Cloud Account Number:	a-t-1100-5350-9900
Current Period:	Dec 11, 2023 to Dec 27, 2025
Expiration Date:	Dec 27, 2025
Repo Id:	4mLnEdi6iu
Site Name:	EnfocusReview

3.1.3. Site Name (siteName)

A site is a place where everything with respect to the assets (folders and files) and themes data is stored. By default, the name of the site would be “EnfocusReview”.

You can find your Site Name in the General Settings of the Enfocus Admin Area. The Site Name is also present in the URL of your Enfocus Review Admin area, for example: <https://portal.eu.enfocus.cloud/review/988nEdi6iuYde6/EnfocusReview/reviews>.



3.1.4. File ID (fileId)

This is the node ID of the node that was created for the document to be uploaded to. The format is as follows : <repositoryId>-<index>.

3.1.5. Presigned URLs

A **presigned URL** is a secure, time-limited URL that allows users to access a specific resource, such as an object in cloud storage like Amazon S3, without needing direct authentication. These URLs are typically used to grant temporary access to resources, allowing users to upload or download files securely without exposing sensitive credentials.

3.1.6. Authentication of requests

The authentication of all the API calls is done via a request header with the name “API-Token”, the value of which is the secure API token received from Enfocus.

3.2. Endpoints

3.2.1. Upload files

It's important to understand that uploading files consists of the following 3 steps:

1. Create a document node and generate a pre-signed URL to upload the file.
2. Upload the file to the generated *presigned URL*.
3. Set the properties to the file.



Note: There are currently 3 ways of uploading files using the Workato platform, but only the single upload API (multipart upload) mentioned below should be used. The other methods are deprecated!

Method	API name	Endpoint	Remarks
Multipart upload	1. Generate presigned URLs for file upload on page 10 2. Set file properties (multipart) on page 12	1. <code>/api/v1/{repold}/{siteName}/files/presignedUrls</code> 2. <code>/api/v1/{repold}/{siteName}/files/{fileId}/properties</code>	This is the preferred way to upload files. You can specify the number of parts for your file; for each part, you'll get a pre-signed URL. You can then upload the files. There is no problem with size here. You can upload files of any size.

3.2.1.1. Multipart upload

The multipart upload method has the following advantages:

- Resilience and fault tolerance:

If a single part upload fails, only that part needs to be retried, rather than restarting the entire upload. This is especially helpful when uploading large files.

- Parallel upload for faster transfer:

Different parts of the file can be uploaded in parallel, significantly improving upload speed compared to a single large file transfer.

- Resuming an upload:

If an upload is interrupted (due to network failure, timeout, etc.), you can resume from the last successful part instead of restarting from scratch.

- Handling large files (beyond 2 GB):

This method is able to handle large files.

1. Generate presigned URLs for file upload

Description

This call is the first step in the file upload process, i.e. the creation of a document node (or collection node as well if required) and the generation of a presigned URL.

Request type

POST

API path

api/v1/{*repold*}/{*siteName*}/files/*presignedURL*

Body

You have to define the 'numberOfParts'. The default is 1.

```
{
  "collection": true,
  "collectionName": "string",
  "preflightReport": true,
  "numberOfParts": 1
}
```

Option	Type	Description
collection	Boolean	This variable specifies whether or not the file that is uploaded is part of a collection.
collectionName	String	The name of the collection to which the file belongs.
preflightReport	Boolean	This variable specifies whether or not the file has a Preflight Report that needs to be attached.
numberOfParts	Integer	Number of parts in your upload. Parts must be at least 5 MB (except for the last part), there can be 1 to 10 000 parts and one part can be at most 5 GB.

Sample response

This is a sample response with 3 parts.

```
{
  "fileId": "4mLnEchHpUP5nj-CNoQPMisWNW",
  "document": {
    "contentUri": [
      "https://tf-test-ga-eu-1-repo20190122160213404400000001.s3-
      accelerate.amazonaws.com/4mLnEchHpUP5nj-CNoQPMisWNW/content/content/1?
      uploadId=dqWy8eSFxqt3MiQzUZoC1bJRzAxSAiww9pMmDka.AjHwQ4D",
      "https://tf-test-ga-eu-1-repo20190122160213404400000001.s3-
      accelerate.amazonaws.com/4mLnEchHpUP5nj-CNoQPMisWNW/content/content/1?
      uploadId=dqWy8eSFxqt3MiQzUZoC1bJRzAxSAiww9pMmDka.AjHwQ4D",
      "https://tf-test-ga-eu-1-repo20190122160213404400000001.s3-
      accelerate.amazonaws.com/4mLnEchHpUP5nj-CNoQPMisWNW/content/content/1?
      uploadId=dqWy8eSFxqt3MiQzUZoC1bJRzAxSAiww9pMmDka.AjHwQ4D"
    ],
    "uploadId": "dqWy8eSFxqt3MiQzUZoC1bJRzAxSAiww9pMmDka.AjHwQ4D_vs-
    iz4WeuBBMz0Mg4PRjmlvnKbV51VlruXqe7h39xQO1sP1DNxhrcQ7zXgV_sLZOq7NQiwM_hl0I.X1dXe-
    cDHvTVrJkJ15Em9HnLqeCv-bP1EEAg5_.yJi20DfU-",
    "statusURI": "CONTENT/STATUS/v0/4mLnEchHpUP5nj-CNoQPMisWNW/content/1?
    multipart=true&uploadId=dqWy8eSFxqt3MiQzUZoC1bJRzAxSAiww9pMmDka.AjHwQ4D_vs-
    iz4WeuBBMz0Mg4PRjmlvnKbV51VlruXqe7h39xQO1sP1DNxhrcQ7zXgV_sLZOq7NQiwM_hl0I.X1dXe-
    cDHvTVrJkJ15Em9HnLqeCv-bP1EEAg5_.yJi20DfU-"
  },
  "preflightReport": {}
}
```

2. Upload content (multipart)**Description**

This is the second step in the upload process. Once you receive all the presigned URLs for the various parts, you need to make a PUT call to each of the presigned URLs with its corresponding part. The upload ID and the part number are embedded in these links.

This is the second step in the upload process. The content (file and Preflight Report) needs to be uploaded to the presigned URL via a PUT call.

URL

These are the pre-signed URLs received in the previous call.

Request type

PUT

Request body

Break the file into multiple chunks and then upload each chunk to its corresponding pre-signed URL.

Sample response

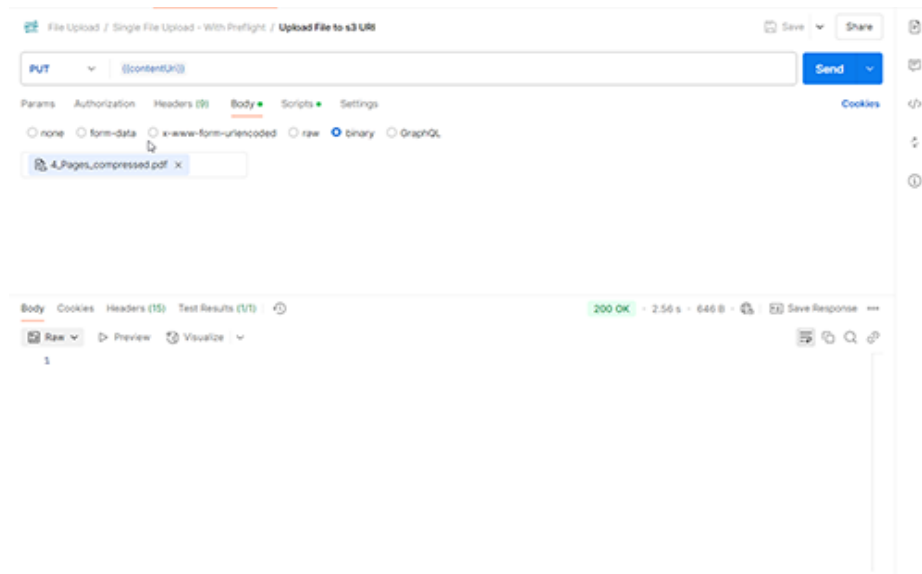
200 OK



Note: If the file has a Preflight Report, make a PUT call with the Preflight Report as the binary data to the pre-signed URL received under “preflightReport” > “contentURI”.

Postman sample

Screenshot of a Postman request to make a PUT call to a presigned URL.



Response header

After successfully making each PUT call, we need to keep a note of the value of the response header "**ETag**". These ETags are necessary to finalize the upload when setting the properties in the next call.

3. Set file properties (multipart)

Description

This is the third step in the file upload process. The review properties have to be set once the file upload to the pre-signed URLs is completed.

Request type

POST

API path

api/v1/{repold}/{siteName}/files/{fileId}/properties



Note: The fileId is part of the response received from the call to [get the presigned URLs](#).

Body

There is one extra category (4. Upload) that is only relevant in case of multipart upload.

1. Connections: At least one connection is required.

Option	Type	Description
name	String	Name of the decision button that will be displayed on the Approval tab in Review Viewer.
color	String	Color of the approval option. Possible values are: • Gray • Yellow • Cyan • Blue • Magenta • Orange • Green • The hexadecimal value of a color, in the format #RRGGBB
commentEnabled	Boolean	When set to true, the reviewer is requested to give a comment when clicking the decision button.
jobWithComments	Boolean	When set to false, the corresponding button will be disabled if there are any annotations added by the user on that document. The reviewer can either remove the annotations or choose a different connection.
translation	JSON	Translated name of the connection, which will be displayed on the corresponding button on the Approval tab in the given language.

2. General options:

Options	Type	Description
originalName	String	Name of the file to be uploaded to Enfocus review. This name will be used to display the file in the viewer, in the job collection and in the Review admin environment.
dueTime	String	This is in the ISO 8601 format for a UTC timestamp . This is the date that will be used to show the due date in the Review admin environment. Ex : 2025-03-10T09:56:54.012Z
deletionDelay	Integer	The number of days after the dueTime when the file should be deleted.
brandNodeid	String	The node ID of the brand the file is associated with.

3. View:

Option	Type	Description
comments	Boolean	If disabled, the annotations widget and the radio button to display the annotations will not be shown in the Review viewer.
doublePageInitialView	Boolean	If enabled, the viewer loads in the double page view (spread view) by default.
doublePageView	Boolean	If enabled, it displays the option to toggle between the spread view and single page view on the Review viewer.
finishedSize	Boolean	If disabled, the toggle to show the finished size of the document is not available.
tools	Boolean	If disabled, the tools tab and a few tools are not displayed in the viewer.
inkCoverage	Boolean	If disabled, the reviewer will not see the ink coverage widget and tool.
layers	Boolean	If disabled, the layers widget will not be displayed in the viewer.
pageBoxes	Boolean	If disabled, the toggle for displaying the page boxes is not displayed in the viewer.
preflightReport	Boolean	If disabled, the preflight report widget is not displayed.
rulers	Boolean	If disabled, the reviewer will not have the toggle to enable or disable the rulers.
viewRotation	Boolean	If disabled, the reviewer will not have the option to rotate the view.
wireframe	Boolean	If disabled, the reviewer will not have the toggle to enable or disable the wireframe.
defaultUnit	String	The default unit used to display the various values in the review viewer. The valid options are : ["p","in","ft","mm","cm","m","pt"]
allowDownload	Boolean	If disabled, the reviewer will not have the option to download the PDF file to a local folder.
coverPage	String	This specified how the coverpage is shown in the review viewer. The valid options are : ["NoCoverPage", "CoverPage", "Sheet", "CoverSpread", "CoverAndInsideCoverSpread"]

4. Upload:

Option	Type	Description
eTags	Array of Strings	In Amazon S3, an ETag (Entity Tag) is a hash (typically an MD5 checksum) used to uniquely

Option	Type	Description
		identify a specific version of an object. When performing a multi-part upload, each uploaded part returns an ETag in the response headers. The Etags that were received in the response headers after the upload of each part needs to be provided in this array.
statusURI	String	This is nothing but the API endpoint to call to update the status of the upload. This is received in the response of the call to get the presigned URLs. This needs to be provided here.

Sample properties of the body

```
{
  "connections": [
    {
      "name": "string",
      "color": "string",
      "commentEnabled": boolean,
      "approverNameEnabled": boolean,
      "jobWithComments": boolean,
      "translations": {
        "da-DK": "string",
        "de-DE": "string",
        "en-US": "string",
        "es-ES": "string",
        "et-EE": "string",
        "fi-FI": "string",
        "fr-FR": "string",
        "it-IT": "string",
        "ja-JP": "string",
        "lt-LT": "string",
        "lv-LV": "string",
        "nb-NO": "string",
        "nl-NL": "string",
        "pl-PL": "string",
        "pt-BR": "string",
        "sv-SE": "string",
        "zh-Hans": "string"
      }
    }
  ],
  "originalName": "string",
  "dueTime": "2025-03-10T09:56:54.012Z",
  "deletionDelay": int,
  "brandNodeId": "string",
  "view": {
    "comments": boolean,
    "doublePageInitialView": boolean,
    "doublePageView": boolean,
    "finishedSize": boolean,
    "tools": boolean,
    "inkCoverage": boolean,
    "layers": boolean,
    "pageBoxes": boolean,
    "preflightReport": boolean,
    "rulers": boolean,
    "viewRotation": boolean,
    "wireframe": boolean,
    "defaultUnit": "string", // One of ["p", "in", "ft", "mm", "cm", "m", "pt"]
    "allowDownload": boolean,
    "coverPage": "string" // One of ["NoCoverPage", "CoverPage", "Sheet",
    "CoverSpread", "CoverAndInsideCoverSpread"]
  },
}
```

```

"upload": {
  "eTags": [
    "string"
  ],
  "statusURI": "string"
}
}

```

Sample Response

In the final response we get the sharedLink and a fileId.

The sharedlink is nothing but the anonymous link to the job collection (if specified while generating the pre-signed url) or the file in the Review viewer.

There is no difference between single part and multipart upload.

```

{
  "sharedlink": "https://token-api.ci.dev.enfocus.esko.rocks/link/v1/6iRfXdJzxGqob7naGwLigQ5gEkpDK11PvBqR6rkEfGsJzRAwwJC3p7ysRiiDpoB8XZRR1Pk3fzej95hHc8yYuBG6pogeaFToBgRsrXGjTJZ6DpVmyL379DXs8yEdQmMcY9N",
  "fileid": "4mLnEdK4hCMp2e-3bF1pQYH8T"
}

```

3.2.2. Delete files

Description

This call is used to delete files and job collections from Enfocus Review.

Request type

DELETE

API path

api/v1/{*repold*}/{*siteName*}/files

Query parameter

Use the *id* parameter to indicate the files that must be deleted. Use a comma to separate multiple files if applicable.

Example: ids=4mLnEchHpUP5nj-CNoSLU1Vh85,4mLnEchHpUP5nj-CNoSMSOzaEV

Sample URL

{{baseURL}}/{{repold}}/{{siteName}}/files?ids=4mLnEchHpUP5nj-CNoSLU1Vh85,4mLnEchHpUP5nj-CNoSMSOzaEV

Sample response

```

{
  "message": "All of the files were successfully deleted."
}

```

```
}
```

3.2.3. Retrieve Review Public API Info

Description

This returns the exact version of the Enfocus Review application.

Request type

GET

API path

api/v1/info

Sample response

```
{
  "productName": "Enfocus Review",
  "version": "2517.2b091d7.2025-03-12t03-23-01z"
}
```

3.2.4. Get Annotations

Description

This call is used to get all the annotations that are available in a document with all details. This is actually an API call from the view service that we are using to provide information about the annotations.

Request type

GET

API path

api/v1/files/[fileId](#)/annotations



Note: Here the `nodePath` refers to the node ID of the document for which we request the annotations.

Sample response

```
{
  "annotations": [
    {
      "annotType": "text",
      "color": "#FFFF00",
      "creationdate": "D:20250313212848+05'30'",
      "date": "D:20250313212848+05'30'",
      "firstname": "",
      "flags": "print",

```

```
    "fullMessageNumber": "2",
    "inreplyto": "",
    "is3D": "false",
    "lastname": "",
    "messageNumber": 2.0,
    "name": "hCMd7rhUpEBaHH1n1QVl62",
    "page": 1.0,
    "popup": {
      "open": "no",
      "page": 0.0,
      "rect":
"74.74960629921262,549.9212598425197,327.03307086614177,677.48031496063"
    },
    "rect": "74.74960629921262,677.48031496063,74.74960629921262,677.48031496063",
    "subject": "Note",
    "username": "PRB"
  }
}
```

3.2.5. Create brand - With logo

Description

This call is used to create a brand with a logo. The request body must contain both brand properties and the logo file.

Request type

POST

API path

api/v1/{*repold*}/{*siteName*}/brands

Request body

Multipart Form Data:

Key	Type	Description
brandProperties	Text	JSON containing the brand properties. More info: Brand properties .
originalFileName	Text	Name of the logo file (must include the correct extension).
logo	File	The actual logo file.

Sample response

```
{
  "message": "Brand added successfully",
  "brandId": "4mLnEdK4hCMP2e-3bFCRwJeE1"
```

```
}
```

3.2.6. Create brand - Without logo

Description

This call is used to create a brand without uploading a logo. The brand properties need to be provided in the request body as multipart/form-data.

Request type

POST

API path

api/v1/{[repold](#)}/{[siteName](#)}/brands

Request body

Multipart Form Data:

Key	Type	Description
brandProperties	Text	JSON containing the brand properties. More info: Brand properties .

Sample response

```
{
  "message": "Brand added successfully",
  "brandId": "4mLnEdK4hCMp2e-3bFCRwJeE1"
}
```

3.2.7. Delete the given brand

Description

This call is used to delete brands from Enfocus Review.

Request type

DELETE

API path

api/v1/{[repold](#)}/{[siteName](#)}/brands

Query parameter

Use the *brandIds* parameter to indicate the brands that must be deleted. Use an ampersand to separate multiple brands if applicable.

Example: brandIds=4mLnEchHpUP5nj-CNoSLU1Vh85&brandIds=4mLnEchHpUP5nj-CNoSMSOzaEV

Sample URL

{{baseUrl}}/deleteBrand/{{repoID}}/{{siteName}}?brandIds=4mLnEchHpUP5nj-CNoSLU1Vh85&brandIds=4mLnEchHpUP5nj-CNoSMSOzaEV

Sample response

```
{
  "failedBrandIds": [],
  "message": "All of the brands were successfully deleted."
}
```

3.2.8. Set the properties of the brand

Description

In order to enable the addition and updating of brands an API recipe has been created on Workato which makes it possible to do all these actions using a single API endpoint.

Request type

POST

API path

api/v1/{*repold*}/{*siteName*}/brands

Body

The body for this API call needs to be supplied as multipart/form-data.

Key	Type	Description
brandProperties	Text	Refer to the Brand Properties section below.
originalFileName	Text	The name of the logo file. Example: Test-Logo.png  Note: Do not forget to add the extension of the file, because we need this to be able to display/process the file.
brandId	Text	The node ID of the brand to be updated.
logo	File	The actual file to be uploaded as a logo.

Brand properties

Property	Type	Description
siteName	string	The title of the webpage when displaying the review viewer or job collection.
appearance	object	Defines appearance settings, including themes and colors.
appearance.dark	object	Dark theme color settings.
appearance.dark.primary	string	Primary color for dark mode. This needs to be a valid Hex color.
appearance.dark.secondary	string	Secondary color for dark mode. This needs to be a valid Hex color.
appearance.dark.text	string	Text color for dark mode. This needs to be a valid Hex color.
appearance.light	object	Light theme color settings.
appearance.light.primary	string	Primary color for light mode. This needs to be a valid Hex color.
appearance.light.secondary	string	Secondary color for light mode. This needs to be a valid Hex color.
appearance.light.text	string	Text color for light mode. This needs to be a valid Hex color.
appearance.theme	string	Specifies the overall theme. The viewer or job collection will be loaded with this theme by default. It can either be "DARK" or "LIGHT".
collections	object	Defines all the collection related help strings.
collections.reviewedFilesHelp	string	Help text or description that is displayed in the job collections UI when all of the files in the job collection are reviewed.
collections.jobCollectionHelp	string	Help text or description that is displayed in the job collections UI when all or some of the files in the job collection are not reviewed.
viewer	object	Defines all the viewer related help strings.
viewer.qualityReportHelp	string	Help text or description that is displayed in the viewer UI in the Preflight Report widget when there are no preflight issues in the document.
companyInfo	string	Company-related information.

Property	Type	Description
companyInfo.postApprovalPage	string	Full URL to the intended post approval page. A valid example would be : https://www.enfocus.com/en/review-thank-you-for-approving
companyInfo.hasPostApproval	string	Indicates if a post-approval page is enabled.
companyInfo.homePage	string	Full URL to the intended homepage of the company.
description	string	A brief description about the brand.
name	string	The name of the brand

Sample

```
{
  "siteName": "string",
  "appearance": {
    "dark": {
      "primary": "string",
      "secondary": "string",
      "text": "string"
    },
    "light": {
      "text": "string",
      "secondary": "string",
      "primary": "string"
    },
    "theme": "string"
  },
  "collections": {
    "reviewedFilesHelp": "string",
    "jobCollectionHelp": "string"
  },
  "viewer": {
    "qualityReportHelp": "string"
  },
  "companyInfo": {
    "postApprovalPage": "string",
    "hasPostApproval": "string",
    "homePage": "string"
  },
  "description": "string",
  "name": "string"
}
```

3.2.9. Fetch the details of a specific brand

Description

This call is used to fetch either the details of all the brands in an Enfocus Review account or the details about a particular brand.

Request type

GET

API path

api/v1/{repold}/{siteName}/brands

- To list the details of **all brands**, use api/v1/{repold}/{siteName}/brands
- To list the details **of one specific brand**, use api/v1/{repold}/{siteName}/brands/{brandId}

Parameters

Minimal : This parameter can be used to obtain minimal details about the brands instead of a detailed description. By default this parameter is considered false.

Sample response

Listing the details of all brands:

```
{
  "brands": [
    {
      "name": "Default Theme",
      "description": "Default theme to get started.",
      "siteName": "Enfocus Review",
      "companyInfo": {
        "homePage": "https://www.enfocus.com",
        "hasPostApproval": "true",
        "postApprovalPage": "https://www.enfocus.com/review-thank-you-for-
approving"
      },
      "viewer": {},
      "collections": {},
      "appearance": {
        "theme": "BOTH",
        "light": {
          "primary": "#61BCFF",
          "secondary": "#0079D1",
          "text": "#000000"
        },
        "dark": {
          "primary": "#0079D1",
          "secondary": "#61BCFF",
          "text": "#FFFFFF"
        }
      },
      "creationDate": "2024-11-12T10:14:28.017000+00:00",
      "type": "DEFAULT",
      "brandId": "4mLnEchHpUP5nj-CNoAaXFMFLv"
    },
    {
      "name": "Test Theme 1",
      "companyInfo": {
        "homePage": "https://www.atomyx.io",
        "hasPostApproval": "false"
      },
      "viewer": {},
      "collections": {},
      "appearance": {
        "theme": "BOTH",
        "light": {
          "primary": "#61BCFF",
          "secondary": "#0079D1",
          "text": "#000000"
        },
        "dark": {
          "primary": "#0079D1",
          "secondary": "#61BCFF",
          "text": "#FFFFFF"
        }
      },
      "creationDate": "2025-03-13T14:16:29.970000+00:00",
      "type": "CUSTOM",
      "brandId": "4mLnEchHpUP5nj-CNoSQ5rm3kt"
    }
  ]
}
```

```
]
}
```

Listing the details of one specific brand:

```
{
  "name": "Default Theme",
  "description": "Default theme to get started.",
  "siteName": "Enfocus Review",
  "companyInfo": {
    "homePage": "https://www.enfocus.com",
    "hasPostApproval": "true",
    "postApprovalPage": "https://www.enfocus.com/review-thank-you-for-approving"
  },
  "viewer": {},
  "collections": {},
  "appearance": {
    "theme": "BOTH",
    "light": {
      "primary": "#61BCFF",
      "secondary": "#0079D1",
      "text": "#000000"
    },
    "dark": {
      "primary": "#0079D1",
      "secondary": "#61BCFF",
      "text": "#FFFFFF"
    }
  },
  "creationDate": "2024-11-12T10:14:28.017000+00:00",
  "type": "DEFAULT",
  "brandId": "4mLnEchHpUP5nj-CNoAaXFMFLv"
}
```

3.2.10. Get the presigned URL for brand logo

Description

This call is used to get the presigned URL for a brand logo.

Request type

POST

API path

api/v0/{[repold](#)}/{[siteName](#)}/listBrands

Parameters

Sample response

3.2.11. Retrieve and list file information - Fetch file details

Description

This call is used to list the information about files that are available in Enfocus Review.



Note: Are there any rate limits set for polling the file status? It is recommended and sufficient to poll once per minute to check for approval.

Request type

GET

API path

api/v1/{*repold*}/{*siteName*}/files

Body

```
{
  "fileIds": [
    "string"
  ]
}
```

Parameters

Key	Type	Description
reviewed	Boolean	This indicates whether or not the PDF has been reviewed.
fileId	String	The node ID of the PDF.
fileName	String	The readable name of the PDF.
connection	String	The approval option that was chosen during the review. Only available for reviewed files.
downloadURL	String	The URL to download the PDF.  Note: This URL is timebound and will expire.
comments	String	The approval comments that were provided during the review. Only available for reviewed files.
found	Boolean	This indicates whether or not the PDF exists in Enfocus Review.

Sample response

```
{
  "files": [
    {
      "reviewed": false,
      "fileId": "4mLnEchHpUP5nj-CNoDJkJ7rRg",
    }
  ]
}
```

```

    "fileName": "TestFile",
    "downloadURL": "https://tf-test-qa-eu-1-repo20190122160213404400000001.s3-
accelerate.amazonaws.com/4mLnEchHpUP5nj-CNoDJk7rRg/content/content/1?
versionId=ry_li0QE19S0A3XLG5.3OHvaVhldTFR8&response-content-disposition=attachment
%3B%20filename%2A%3DUTF-8%27%27filedc8f8c77a33e469cb11660e2a6a4f63f.pdf&X-
Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20250313T150703Z&X-Amz-
SignedHeaders=host&X-Amz-Expires=3600&X-Amz-Credential=AKIA3KDEWFO7HHVXCEHC
%2F20250313%2Feu-west-1%2Fs3%2Faws4_request&X-Amz-
Signature=332824cbbfd40f3b658ad52c5faac4adac7a9547c720e55ffd6f0f4051db255a",
    "found": true
  },
  {
    "reviewed": true,
    "fileId": "4mLnEchHpUP5nj-CNoSNyqkicB",
    "fileName": "Producer.pdf",
    "connection": "Test",
    "downloadURL": "https://tf-test-qa-eu-1-repo20190122160213404400000001.s3-
accelerate.amazonaws.com/4mLnEchHpUP5nj-CNoSNyqkicB/content/content/1?
versionId=P9vCexqjN4lSo2s7s3A6kBEHdVv..sl3&response-content-disposition=attachment
%3B%20filename%2A%3DUTF-8%27%27file23a61af4da524dd1bb3f580b88f93f78.pdf&X-
Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Date=20250313T150703Z&X-Amz-
SignedHeaders=host&X-Amz-Expires=3600&X-Amz-Credential=AKIA3KDEWFO7HHVXCEHC
%2F20250313%2Feu-west-1%2Fs3%2Faws4_request&X-Amz-
Signature=0c1dac942c3871ea2deba3e7e2e2b1c73c524da57fc5b811abc4cc4faeb4a688",
    "comments": "",
    "found": true
  }
]
}

```

3.2.12. Get a list of all the available collections in the given account

Description

This call is used to give an overview of all the job collections in the Enfocus Review application.

Request type

GET

API path

api/v1/{*repold*}/{*siteName*}/collections

Sample response

```

{
  "collections": [
    {
      "jobCollectionId": "4mLnEchHpUP5nj-CNoFqdg3HKN",
      "jobCollectionTitle": "Test_Collection_30JAN",
      "jobCollectionURL": "https://review.qa-eu-1.test.enfocus.esko.rocks/
review/4mLnEchHpUP5nj/PRB_WORKATO/jobcollection?id=4mLnEchHpUP5nj-CNoFqdg3HKN"
    },
    {
      "jobCollectionId": "4mLnEchHpUP5nj-CNoEGhQZiWr",
      "jobCollectionTitle": "15_JAN",
      "jobCollectionURL": "https://review.qa-eu-1.test.enfocus.esko.rocks/
review/4mLnEchHpUP5nj/PRB_WORKATO/jobcollection?id=4mLnEchHpUP5nj-CNoEGhQZiWr"
    },
    {
      "jobCollectionId": "4mLnEchHpUP5nj-CNoFpzrj3xn",
      "jobCollectionTitle": "30_JAN",
    }
  ]
}

```

```

      "jobCollectionURL": "https://review.qa-eu-1.test.enfocus.esko.rocks/
review/4mLnEchHpUP5nj/PRB_WORKATO/jobcollection?id=4mLnEchHpUP5nj-CNoFpzrj3xn"
    },
    {
      "jobCollectionId": "4mLnEchHpUP5nj-CNoDVbczc8i",
      "jobCollectionTitle": "07_JAN",
      "jobCollectionURL": "https://review.qa-eu-1.test.enfocus.esko.rocks/
review/4mLnEchHpUP5nj/PRB_WORKATO/jobcollection?id=4mLnEchHpUP5nj-CNoDVbczc8i"
    },
    {
      "jobCollectionId": "4mLnEchHpUP5nj-CNo9zYi4x5q",
      "jobCollectionTitle": "Test_Collection_11NOV",
      "jobCollectionURL": "https://review.qa-eu-1.test.enfocus.esko.rocks/
review/4mLnEchHpUP5nj/PRB_WORKATO/jobcollection?id=4mLnEchHpUP5nj-CNo9zYi4x5q"
    }
  ]
}

```

3.2.13. Retrieve and list file information - List all files

Description

This call is used to list the files that are available in Enfocus Review. You can use filters to fetch the details of the files.

Request type

GET

API path

api/v1/{*repold*}/{*siteName*}/files

Parameters

There are multiple parameters that can be used in this API to list the documents based on certain conditions. If none of the parameters are used, then all the files - both individual files and files in collections - in the Enfocus Review account are listed.

Parameter	Type	Description
reviewed	Boolean	- If enabled all the reviewed files are returned. - If disabled all the non-reviewed files are returned.  Note: By default this parameter is not included.
brandId	String	The node ID of the brand must be provided. All the files associated with this brand will be returned.
jobCollectionId	String	The node ID of the job collection. All the files in this job collection are returned.
From	Integer	The files after the index that have to be fetched.

Parameter	Type	Description
		Example: If 10 is provided as value, the files from the 10th position onwards in the database will be fetched.
Size	Integer	The number of files from the "From" index that needs to be fetched. Example: If "From" is set to 10 and "Size" is set to 20, the files from 11-30 will be returned.  Note: If "Size" is set to 0, the total number of files in the account is returned.

**Note:**

- It is possible to combine two parameters and fetch the files obeying those conditions. For example, if the user wants all the files that are reviewed and belong to particular brand then they can formulate the request url as follows : `/api/v0/review/list/{repold}/{siteName}?reviewed=true&brandId={{validBrandId}}`
- The reason for providing the from and size parameters is because we internally use elastic search queries to fetch the data. It has a hard limit on returning 10000 entries. So, if the customer has large number of files, it is advisable to use the from and size parameters to obtain the files in batches.

3.2.14. Update the properties of the brand - Both the logo and the brand properties

Description

This page describes how to update both the logo and the properties of a particular brand.

Request type

POST

API path

`api/v1/{repold}/{siteName}/brands`

Request body

The body for this API call needs to be supplied as multipart/form-data.

Key	Type	Description
brandProperties	Text	JSON containing the brand properties. More info: Brand properties .

Key	Type	Description
brandId	Text	The node ID of the brand that needs to be updated.
originalFileName	Text	Name of the new logo file (must include the correct extension).
logo	File	The new logo file.

Sample response

```
{
  "brandId": "4mLnEchHpUP5nj-CNoQyQK8oZi",
  "message": "Brand updated successfully"
}
```

3.2.15. Update the properties of the brand - Without modifying the logo

Description

This page describes how to update the specific brand properties

Request type

POST

API path

api/v1/{[repold](#)}/{[siteName](#)}/brands

Request body

The body for this API call needs to be supplied as multipart/form-data.

Key	Type	Description
brandProperties	Text	JSON containing the brand properties. More info: Brand properties .
brandId	Text	The node ID of the brand to be updated.

Sample response

```
{
  "brandId": "4mLnEchHpUP5nj-CNoQyQK8oZi",
  "message": "Brand updated successfully"
}
```

4. Copyrights

© 2026 Enfocus BV all rights reserved. Enfocus is an Esko company.

Certified PDF is a registered trademark of Enfocus BV.

Enfocus PitStop Pro, Enfocus PitStop Workgroup Manager, Enfocus PitStop Server, Enfocus BoardingPass, Enfocus Connect YOU, Enfocus Connect ALL, Enfocus Connect SEND, Enfocus StatusCheck, Enfocus CertifiedPDF.net, Enfocus PDF Workflow Suite, Enfocus Switch, Enfocus SwitchClient, Enfocus SwitchScripter, Enfocus TestDrive, Enfocus SwitchScriptTool, Enfocus Browser, PitStop Library Container, Enfocus Griffin, Enfocus Review, Enfocus FastLane and Enfocus Appstore are product names of Enfocus BV.

Adobe, Acrobat, Distiller, InDesign, Illustrator, Photoshop, FrameMaker, PDFWriter, PageMaker, Adobe PDF Library™, the Adobe logo, the Acrobat logo and PostScript are trademarks of Adobe Systems Incorporated.

Datalogics, the Datalogics logo, PDF2IMG™ and DLE™ are trademarks of Datalogics, Inc.

Apple, Mac, macOS, Macintosh, iPad and ColorSync are trademarks of Apple Computer, Inc. registered in the U.S. and other countries. Windows and Windows Server are registered trademarks of Microsoft Corporation.

PANTONE® Colors displayed here may not match PANTONE-identified standards. Consult current PANTONE Color Publications for accurate color. PANTONE® and other Pantone, Inc. trademarks are the property of Pantone, Inc. ©Pantone, Inc., 2006.

OPI is a trademark of Aldus Corporation.

Quark, QuarkXPress, QuarkXTensions, XTensions and the XTensions logo among others, are trademarks of Quark, Inc. and all applicable affiliated companies, Reg. U.S. Pat. & Tm. Off. and in many other countries.

Docker and the Docker logo are trademarks or registered trademarks of Docker, Inc. in the United States and/or other countries. Docker, Inc. and other parties may also have trademark rights in other terms used herein.

This product and use of this product is under license from Markzware under U.S. Patent No. 5,963,641.

Other brand and product names may be trademarks or registered trademarks of their respective holders. All specifications, terms and descriptions of products and services are subject to change without notice or recourse.