

Documentation for the Pubhub Ingestion API - ingestion.pubhub.dk

Revision history

Date	Version	Comment
2021-05-28	1	First revision of documentation
2021-06-18	1.1	Info about qa/test environment added
2021-07-21	1.2	“PI-E015: Front cover is not valid - it must be max 50 MB” added
2023-04-21 1	1.3	Endpoint version links updated to v3 (v1 supports Danish ONIX subset 1.0.6, v2 supports v1.0.7 and v3 supports 1.0.8)
2024-01-22	1.4	API v4: Supports Danish ONIX subset 1.1.1
2024-10-01	1.4.1	New validation error codes, PI-C025 and PI-C026, added. Minor change so latest URL version is still v4.
2024-11-18	1.5	Input is validated with updated Dansk onix

The following describes the different REST endpoints that can be used to send in material via the new Pubhub Ingestion Api.

The API supports the new Danish Onix subset, see more at <http://danskonix.dk/om.html>

Test environment: <http://ingestion.qa.pubhub.dk>.

Production environment: <https://ingestion.pubhub.dk>.

The test environment for the Pubhub admin site is: <http://admin.qa.pubhub.dk/>

API Versions

- V1: <https://ingestion.pubhub.dk/api/v1/>: Supports Danish ONIX Subset 1.0.6
- V2: <https://ingestion.pubhub.dk/api/v2/>: Supports Danish ONIX Subset 1.0.7
- V3: <https://ingestion.pubhub.dk/api/v3/>: Supports Danish ONIX Subset 1.0.8
- V4: <https://ingestion.pubhub.dk/api/v4/>: Supports Danish ONIX Subset 1.1.1
- V6: <https://ingestion.pubhub.dk/api/v4/>: Supports Danish ONIX Subset 1.1.3

Order of operations

It is recommended to use the endpoints in the following order:

- 1) Validate Onix / Validate media files
- 2) Upload Onix
- 3) Upload supporting resource
- 4) Upload audio- or ebook media file

5) Upload sample or generate sample

Note that step 1 technically can be omitted since validation is also performed during upload

REST Endpoints Security

All HTTP requests to the Ingestion API must include the x-service-key header which indicates the license key (the key provided in the samples is just a dummy one). Contact Publizon if you do not have a license key.

Endpoints

Validate Onix

Description	This endpoint can be used to validate an Onix file, ie. check if it is valid according to the Danish Onix subset.
Endpoint	<code>https://ingestion.pubhub.dk/api/<API_VERSION>/validate/onix</code>
HTTP verb	POST
Required header	x-service-key, Content-Type
Body	The Onix XML
Example request	<pre>curl --location --request POST 'https://ingestion.pubhub.dk/api/<API_VERSION>/validate/onix' --header 'x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0' --header 'Content-Type: application/xml' -d @9788340959028.xml</pre>
JSON response example	<pre>Success: { "success": true, "errors": [], "products": { "9788340959028": { "success": true, "errors": [], "warnings": [] } } } Error { "success": false, "errors": ["[PI-D001] XSD validation error: Element Website does not satisfy assertion. 'WebsiteRole' is required. (line: 1, col: 4173)"], "products": {} }</pre>

Validate Media

Description	This endpoint can be used to validate a media file (ebook or audiobook) to see if it contains any errors.
Endpoint	<a href="https://ingestion.pubhub.dk/api/<API_VERSION>/validate/media">https://ingestion.pubhub.dk/api/<API_VERSION>/validate/media
HTTP verb	POST
Required header	x-service-key, Content-Type
Body	The media file (.epub,.pdf or .zip)
Example request	<pre>curl --location --request POST 'https://ingestion.pubhub.dk/api/<API_VERSION>/validate/media' --header 'x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0' --header 'Content-Type: application/epub+zip' --data-binary @9788340959028.zip</pre>
Json response example	<pre>Success: { "success": true, "warnings": [], "errors": [] } Error { "success": false, "warnings": [], "errors": ["PI-G003[] Invalid bit rate 128 in file del002.mp3. Only 96 is accepted for CBR files."] }</pre>

Upload Onix

Description	This endpoint can be used to upload Onix, eg. create a new book. The Onix xml must be valid according to the Danish Onix Subset.
Endpoint	<a href="https://ingestion.pubhub.dk/api/<API_VERSION>/onix">https://ingestion.pubhub.dk/api/<API_VERSION>/onix
HTTP verb	POST
Required header	x-service-key, Content-Type
Body	The Onix XML
Example request	<pre>curl --location --request POST 'https://ingestion.pubhub.dk/api/<API_VERSION>/onix' --header 'x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0' --header 'Content-Type: application/xml' -d @9788340959028.xml</pre>
Json response example	<pre>Success: { "success": true, "errors": [],</pre>

	<pre> "products": { "9788340959028": { "success": true, "errors": [], "warnings": [] } } Error { "success": false, "errors": [], "products": { "9788340959028": { "success": false, "errors": ["[PI-A002] Error saving product:\n3Party receiver with id 2219 not found."], "warnings": [] } } } </pre>
--	---

Upload Supporting resource

Description	This endpoint can be used to upload supporting resources to a book, ie. front cover, contributor picture etc. The record reference of the book is supplied as part of the URL, and the resourceContentType is added as a parameter. 01 is front cover, for a full list of options see the Danish Onix Subset definition.
Endpoint	<a href="https://ingestion.pubhub.dk/api/<API_VERSION>/supportingresource/<RECORDREF>?resourceContentType=<TYPE>">https://ingestion.pubhub.dk/api/<API_VERSION>/supportingresource/<RECORDREF>?resourceContentType=<TYPE>
HTTP verb	PUT
Required header	x-service-key, Content-Type
Body	The resource file
Example request	curl --location --request PUT 'https://ingestion.pubhub.dk/api/<API_VERSION>/supportingresource/9788340959028?resourceContentType=01' --header 'x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0' --header 'Content-Type: image/jpeg' --data-binary '@Cover.jpg'
Json response example	<pre> Success: { "success": true, "warnings": [], "errors": [] } Error </pre>

	<pre> { "success": false, "warnings": [], "errors": ["[PI-E014] Front cover is not valid - it must be jpg, jpeg or png and have a width >= 1400 pixels"] } </pre>
--	--

Upload Supporting resource

Description	This endpoint can be used to upload supporting resources to a book, ie. front cover, contributor picture etc. The record reference of the book is supplied as part of the URL, and the resourceContentType is added as a parameter. 01 is front cover, for a full list of options see the Danish Onix Subset definition.
Endpoint	<a href="https://ingestion.pubhub.dk/api/<API_VERSION>/supportingresource/<RECORDREF>?resourceContentType=<TYPE>">https://ingestion.pubhub.dk/api/<API_VERSION>/supportingresource/<RECORDREF>?resourceContentType=<TYPE>
HTTP verb	PUT
Required header	x-service-key, Content-Type
Body	The resource file
Example request	<pre> curl --location --request PUT 'https://ingestion.pubhub.dk/api/<API_VERSION>/supportingresource/9788340959 028?resourceContentType=01' --header 'x-service-key: 61FE627A-7A12-4B49-B70E- 934B262131B0' --header 'Content-Type: image/jpeg' --data-binary '@Cover.jpg' </pre>
Json response example	<pre> Success: { "success": true, "warnings": [], "errors": [] } Error { "success": false, "warnings": [], "errors": ["[PI-E014] Front cover is not valid - it must be jpg, jpeg or png and have a width >= 1400 pixels"] } </pre>

Upload audio- or ebook

Description	This endpoint can be used to upload an ebook or audiobook. The record reference of the book is supplied as part of the URL.
Endpoint	<code>https://ingestion.pubhub.dk/api/<API_VERSION>/media/<RECORDREFERENCE></code>
HTTP verb	PUT
Required header	x-service-key
Body	The ebook or audiobook
Example Request	<code>curl https://ingestion.pubhub.dk/api/<API_VERSION>/media/9788340959028 --upload-file 9788340959028.epub --header "x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0"</code>
Json response example	<pre> Success: { "success": true, "warnings": [], "errors": [] } Error { "success": false, "warnings": [], "errors": ["[PI-G003] Invalid bit rate 128 in file del002.mp3. Only 96 is accepted for CBR files."] } </pre>

Upload sample

Description	This endpoint can be used to upload a sample file. Ebook samples must have file extension .pdf or .epub. Audio file samples must have file extension .mp3. The record reference of the book is supplied as part of the URL.
Endpoint	<code>https://ingestion.pubhub.dk/api/<API_VERSION>/sample/<RECORDREF></code>
HTTP verb	Put
Required header	x-service-key
Body	The sample file
Example request	<code>curl https://ingestion.pubhub.dk/api/<API_VERSION>/sample/9788340959028 --upload-file 9788340959028.mp3 --header "x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0" --verbose</code>
Json response	<pre> Success: { </pre>

example	<pre> "success": true, "warnings": [], "errors": [] } Error { "success": false, "warnings": [], "errors": ["[PI-F002] ERROR(RSC-005): C:/Temp/cde485c3-cb04-4100-a4bc-6f388e6282cd.epub/OEBPS/content.opf(12,79): Error while parsing file: Property \"role\" must refine a \"creator\" or \"contributor\" property.."] } </pre>
---------	--

Generate sample

Description	This endpoint can be used to generate a sample file. Note that this is only supported for ebooks. For epubs and pdf you can specify a parameter: "sampleSizePct" to indicate how large in % the sample should be. Alternatively for pdfs only you can specify two parameters: "sampleStartPage" and "samplePages" to specify which part of the pdf to include in the sample.
Endpoint	https://ingestion.pubhub.dk/api/<API_VERSION>/sample/generate/<RECORDREF>
HTTP verb	POST
Required header	x-service-key, Content-Type
Body	Json specifying either 1) the sampleSizePct parameter OR 2) the sampleStartPage and samplePages. Eg. <pre> { "sampleStartPage": 2, "samplePages": 3 } </pre>
Example request	<pre> curl --location --request POST 'https://ingestion.pubhub.dk/api/<API_VERSION>/sample/generate/9788711583906' --header 'x-service-key: 61FE627A-7A12-4B49-B70E-934B262131B0' --header 'Content-Type: application/json' --data-raw '{ "sampleSizePct": 5 }' </pre>
Json response example	<pre> Success: { "success": true, "warnings": [], "errors": [] } Error </pre>

	<pre> { "success": false, "warnings": [], "errors": ["[PI-A001] Content not found. (Error id: 7783644e-de84-41d1-b580- e1a0e2a3fe85)"] } </pre>
--	---

List of error codes

Error code	Description
PI-A001	Content not found.
PI-A002	Error saving product:
PI-B001	Identifier x is already in use by another supplier.
PI-B002	Identifier x is already in use with a different record reference x.
PI-B003	Unexpected error occurred: x
PI-B004	Invalid ResourceContentType x.
PI-B005	Request contains no data.
PI-C001	x is not not a valid identifier.
PI-C002	PrimaryContentType can not be null.
PI-C003	SalesOutlet IDValue is inactive or invalid x.
PI-C004	SalesRestrictionType 00 and SalesRestrictionNode 2 are defined with SalesRestrictionType 06 or 09.
PI-C005	x is not a valid subject code.
PI-C006	Qualifiers cannot be marked as MainSubject (subject code: x).
PI-C007	Only one MainSubject is allowed.
PI-C008	No valid subject category found.
PI-C009	Notification type x is not supported.
PI-C010	More than one EpubConstraint with EpubUsageType 01 and EpubUsageStatus was found.
PI-C011	Contributor role code A01 (By (author)) is required.
PI-C012	Contributor role code E07 (Read by) is required for audio books.
PI-C013	Contributor role code E07 (Read by) is not valid for e-books.
PI-C014	Contributor x is missing KeyNames value.
PI-C015	Page count is not valid for audio books.
PI-C016	Duration is not valid for e-books.
PI-C017	Price with price type x not found.
PI-C018	TitleElement with role code 01 (Product) not found.
PI-C019	EpubTechnicalProtection not found.
PI-C020	x is not a valid EpubTechnicalProtection value for audio books. Only 02 (Digital watermarking) is valid.

PI-C021	TextContent with TextType 03 (Long description) not found.
PI-C022	Document root not found.
PI-C023	XML namespace is invalid x. Please use a valid ONIX namespace x.
PI-C024	XML root node name is invalid x. Please use a valid ONIX root node name x.
PI-C025	A loan-based SalesOutlet was defined together with SalesRestrictionType 09.
PI-C026	A sale-based SalesOutlet was defined together with SalesRestrictionType 06.
PI-D001	XSD validation error: x
PI-D003	Error parsing XML: x
PI-D004	The XSD type x could not be found.
PI-E001	The detected file format x does not match the existing file format x.
PI-E002	The detected file format x does not match the content's file format x.
PI-E003	The image format is not supported. Only JPG and PNG is supported.
PI-E004	Samples can only be generated for e-books.
PI-E005	Both sample start page and number of sample pages are required if percentage is not defined. PI-E006 For EPUB sample generation you must specify a percentage.
PI-E007	Sample generation parameters missing.
PI-E008	The file could not be uploaded to Adobe Content Server: x.
PI-E009	EPUB sample generation error: x.
PI-E010	PDF sample generation error: x.
PI-E011	Unsupported file type. File header: x.
PI-E012	Unknown file type. Not enough data.
PI-E013	Media files cannot be uploaded for physical books.
PI-E014	Front cover is not valid - it must be jpg
PI-E015	Front cover is not valid - it must be max 50 MB
PI-F001	The e-book is watermarked. Please upload a file which is not watermarked.
PI-F003	A stream version of the e-book could not be created. The book has been uploaded and will be available for sale and/or loan
PI-F004	A stream version of the e-book sample could not be created. The book sample has been created and will be available for sale and/or loan
PI-G001	Invalid file x. The zip file must only contain mp3 files.
PI-G002	The file x was not expected. Expected was: y.
PI-G003	Invalid bit rate x in file y. Only 96 is accepted.
PI-G004	Invalid sample rate x in file y. Only 44100 is accepted.
PI-G005	Audio file must be mono but has x audio channels.
PI-G006	No ID3v2 tag found in x.
PI-G007	No metadata found in x.
PI-G008	No title found in ID3v2 tag in file x.
PI-G009	No album found in ID3v2 tag in file x.

PI-G010	No artist found in ID3v2 tag in file x.
PI-G011	No genre found in ID3v2 tag in file x.
PI-G012	No comment found in ID3v2 tag in file x.
PI-G013	No year found in ID3v2 tag in file x.
PI-G014	No track found in ID3v2 tag in file x.
PI-G015	No picture found in file x.
PI-G016	The zip file must not contain subdirectories.
PI-G017	The zip file does not contain any files (possibly because it could not be extracted).