



NewgenONE Content Cloud

Developer Guide

Version: 2024.1

Disclaimer

This document contains information proprietary to Newgen Software Technologies Ltd. User may not disclose or use any proprietary information or use any part of this document without written permission from Newgen Software Technologies Ltd.

Newgen Software Technologies Ltd. makes no representations or warranties regarding any software or to the contents or use of this guide. It also specifically disclaims any express or implied warranties of merchantability, title, or fitness for any particular purpose. Even though Newgen Software Technologies Ltd. has tested the hardware and software and reviewed the documentation, it does not guarantee or imply that this document is error free or accurate regarding any particular specification. As a result, this product is sold as it is and user, the purchaser, is assuming the entire risk as to its quality and performance. Further, Newgen Software Technologies Ltd. reserves the right to revise this publication and make changes in its content without any obligation to notify any person, of such revisions or changes. Newgen Software Technologies Ltd. authorizes no Newgen agent, dealer or employee to make any modification, extension, or addition to the above statements.

Newgen Software Technologies Ltd. has attempted to supply trademark information about company names, products, and services mentioned in this document. Trademarks indicated below were derived from various sources.

Copyright © 2024 **Newgen Software Technologies Ltd.** All Rights Reserved.
No part of this publication may be reproduced and distributed without the prior permission of Newgen Software Technologies Ltd.

Newgen Software, Registered Office, New Delhi

E-44/13

Okhla Phase - II

New Delhi 110020

India

Phone: +91 1146 533 200

info@newgensoft.com

Contents

Preface	10
Revision history	10
About this guide.....	10
Intended audience	10
Related documents	11
Documentation feedback	11
Getting started	12
Registering a new tenant account.....	12
Registering an application.....	13
Microservices	17
App services.....	17
Headers.....	18
Code API.....	21
Headers	21
Sample request.....	22
Sample response	23
Token API.....	23
Headers	24
Sample request.....	25
Sample response	26
Getting app by ID.....	27
Sample request.....	27
Sample response	28
Getting app list	29
Sample request.....	29
Sample response	30
Generating auth code.....	33
Sample request.....	33
Sample response	35
Generating access token	35
Sample request.....	35
Sample response	40
Validating access token.....	41
Sample request.....	41
Sample response	42
Deleting access token	43
Sample request.....	43
Sample response	44
Error codes.....	44
User services.....	45

Headers.....	46
Getting users list.....	49
Sample request.....	49
Sample response	50
Getting user details	52
Sample request.....	52
Sample response	53
Error codes.....	54
Role services.....	56
Headers.....	57
Getting role by ID.....	60
Sample request.....	60
Sample response	61
Getting all roles.....	62
Sample request.....	63
Sample response	64
Getting roles for multiple user IDs.....	66
Sample request.....	66
Sample response	68
Getting multiple roles by role IDs.....	70
Sample request.....	70
Sample response	72
Error codes.....	74
Folder services	76
Headers.....	77
Adding folder	79
Sample request.....	79
Sample response.....	82
Adding folders in bulk	84
Sample request.....	84
Sample response	88
Modifying folder properties	89
Sample request.....	90
Sample response	91
Getting list of all folders.....	94
Sample response	95
Getting folder root path by ID.....	98
Sample request.....	99
Sample response	99
Getting list of all immediate child elements for a parent folder.....	103
Sample request.....	103
Sample response	104
Getting a folder metadata by ID.....	107
Sample request.....	107
Getting a folder data class by ID.....	108
Sample request.....	108

Sample response	109
Getting a folder by ID	109
Sample request.....	109
Sample response	110
Copying a folder.....	112
Sample request.....	112
Sample response	113
Checking copy folder status.....	115
Sample request.....	115
Sample response	116
Moving a folder to trash	116
Sample request.....	116
Sample response	117
Moving a folder from trash.....	120
Sample request.....	121
Sample response	121
Moving a folder.....	123
Sample request.....	124
Sample response	124
Searching folder on data class.....	126
Sample request.....	126
Sample response	127
Search folders based on metadata.....	129
Sample request.....	130
Sample response	130
Deleting a folder containing children	132
Sample request.....	133
Sample response	133
Deleting a folder	134
Sample request.....	134
Sample response	134
Deleting a folder data class by ID.....	135
Sample request.....	135
Sample response	135
Deleting a folder metadata by ID.....	136
Sample request.....	136
Sample response	136
Error codes.....	137
Content services.....	138
Headers.....	140
Moving content to target folder	141
Sample request.....	142
Sample response	142
Copying content.....	146
Sample request.....	146
Sample response	147

Searching content on metadata.....	150
Sample request.....	150
Sample response	151
Checking-in content sync.....	154
Sample request.....	155
Sample response	160
Getting content metadata by ID.....	163
Sample request.....	163
Sample response	164
Getting content data class by ID.....	164
Sample request.....	165
Sample response	165
Getting content root path by ID.....	166
Sample request.....	166
Sample response	167
Sorting content based on revised date time	171
Sample request.....	172
Sample response	172
Searching content based on data class	176
Sample request.....	176
Sample response	177
Searching for largest version	181
Sample request.....	181
Sample response	182
Uploading a file.....	182
Generating Blob URI.....	182
Sample request.....	183
Sample response.....	185
Uploading to Blob.....	188
Sample request.....	188
Getting all content	189
Sample request.....	189
Sample response	191
Getting content status.....	195
Sample request.....	195
Sample response	196
Retrieving content	196
Sample request.....	197
Sample response	197
Fetching content properties.....	198
Sample request.....	198
Sample response	199
Deleting content	202
Sample request.....	203
Sample response	203
Deleting content by parent.....	203

Sample request.....	204
Sample response	204
Deleting content metadata by ID	204
Sample request.....	205
Sample response	205
Deleting content data class by ID	205
Sample request.....	206
Sample response	206
Moving content from trash.....	206
Sample request.....	207
Sample response	208
Marking content to trash.....	212
Sample request.....	212
Sample response	213
Updating parent folder with respect to trash content.....	217
Sample request.....	217
Sample response	218
Moving content to trash.....	221
Sample request.....	222
Sample response	222
Modifying content properties	227
Sample request.....	227
Sample response	229
Content checked out	232
Sample request.....	233
Sample response	233
Setting content version as latest.....	237
Sample request.....	237
Sample response	238
Content undo checked out.....	241
Sample request.....	242
Sample response	242
Error codes.....	245
Secret key services.....	247
Headers.....	247
Getting a secret key	249
Sample request.....	249
Sample response	250
Getting multiple secret keys.....	251
Sample request.....	251
Sample response	252
Getting all secret keys.....	253
Sample request.....	253
Sample response	254
Validating secret key	255
Sample request.....	255

Sample response	255
Error codes.....	256
Security classification services.....	257
Headers.....	258
Getting list of all security classes.....	260
Sample response	260
Getting list of all global tags.....	261
Sample response	261
Error codes.....	262
Data class services	263
Headers.....	263
Getting data class by ID.....	265
Sample request.....	265
Sample response	265
Getting data class by name	268
Sample request.....	268
Sample response	268
Searching data class.....	270
Sample request.....	271
Sample response	271
Error codes.....	273
Notes services	274
Headers.....	275
Creating notes.....	276
Sample request.....	277
Sample response	278
Getting notes by notes ID	279
Sample request.....	280
Sample response	280
Getting notes by object ID	281
Sample request.....	282
Sample response	282
Modifying notes by notes ID	283
Sample request.....	284
Sample response	285
Deleting notes.....	286
Sample request.....	286
Sample response	287
Error codes.....	287
Search service	288
Headers.....	288
Searching files and folders	291
Sample request.....	291
Sample response	294
Annotation services.....	297
Headers.....	297

Saving an annotation	299
Sample request.....	299
Sample response	301
Getting annotation data	301
Sample Request.....	302
Sample response	303
Document download service	304
Headers.....	304
Downloading documents.....	306
Sample request.....	306
Sample response	307
Sample code	308
Common error codes.....	312

Preface

This chapter provides information about the purpose of this guide, details on the intended audience, revision history, and related documents for NewgenONE Content Cloud.

Revision history

Revision date	Description
June 2024	Initial publication

About this guide

This developer guide explains how clients or end-users can integrate their products or applications with NewgenONE Content Cloud.

Intended audience

This developer guide is intended for internal and external product implementation teams responsible for integrating NewgenONE Content Cloud Microservices with other products or platforms. The reader must have an understanding of API Signature formats, specifically JSON, in order to comprehend API requests and responses. Administrative rights might be required to perform the integration operations.

To ensure you are referring to the latest and most recent revision of this guide, download it from one of the following locations:



- [Newgen Internal Doc Portal](#), if you are a Newgen employee.
- [Newgen Partner Portal](#), if you are a Newgen partner.

Related documents

The following documents are related to Newgen Content Cloud:

- NewgenONE Content Cloud Administration Guide
- NewgenONE Content Cloud User Guide for Micro UI

Documentation feedback

To provide feedback or any improvement suggestions on technical documentation, write an email to docs.feedback@newgensoft.com.

To help capture your feedback effectively, share the following information in your email:

- Document name
- Version
- Chapter, topic, or section
- Feedback or suggestions

Getting started

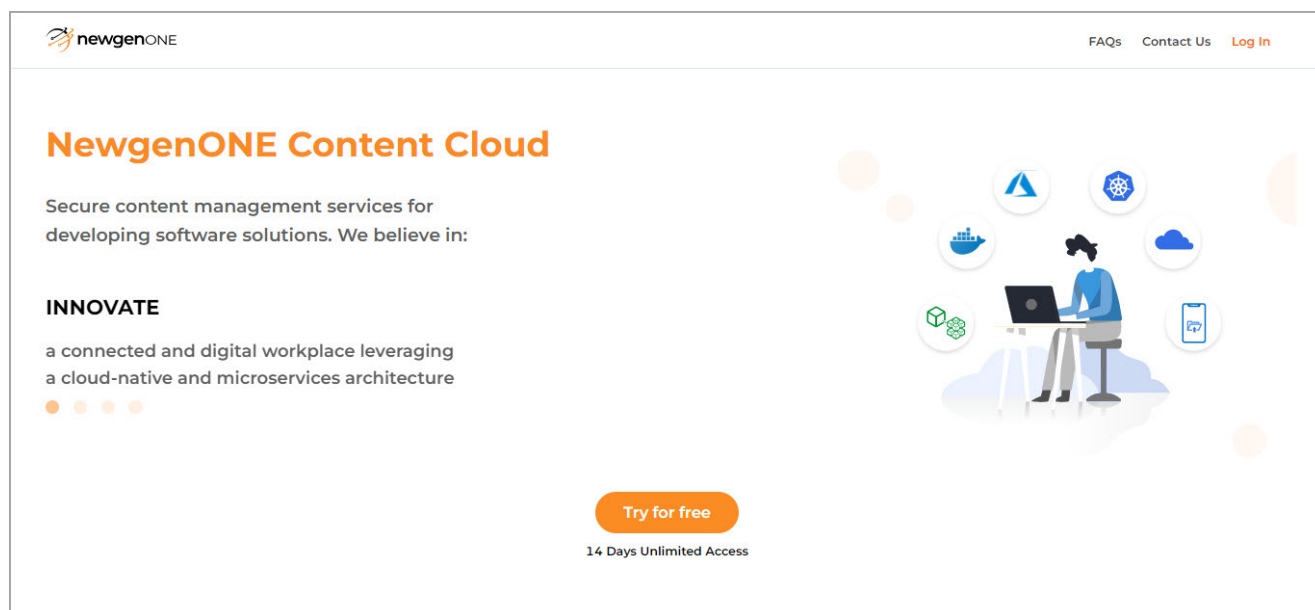
This section explains how to register a [new tenant account](#) and an [application](#).

Registering a new tenant account

NewgenONE Content Cloud allows you to register a new tenant account that provides you access to the microservices-based REST APIs content services, Micro UI, and API Guide & Try-out.

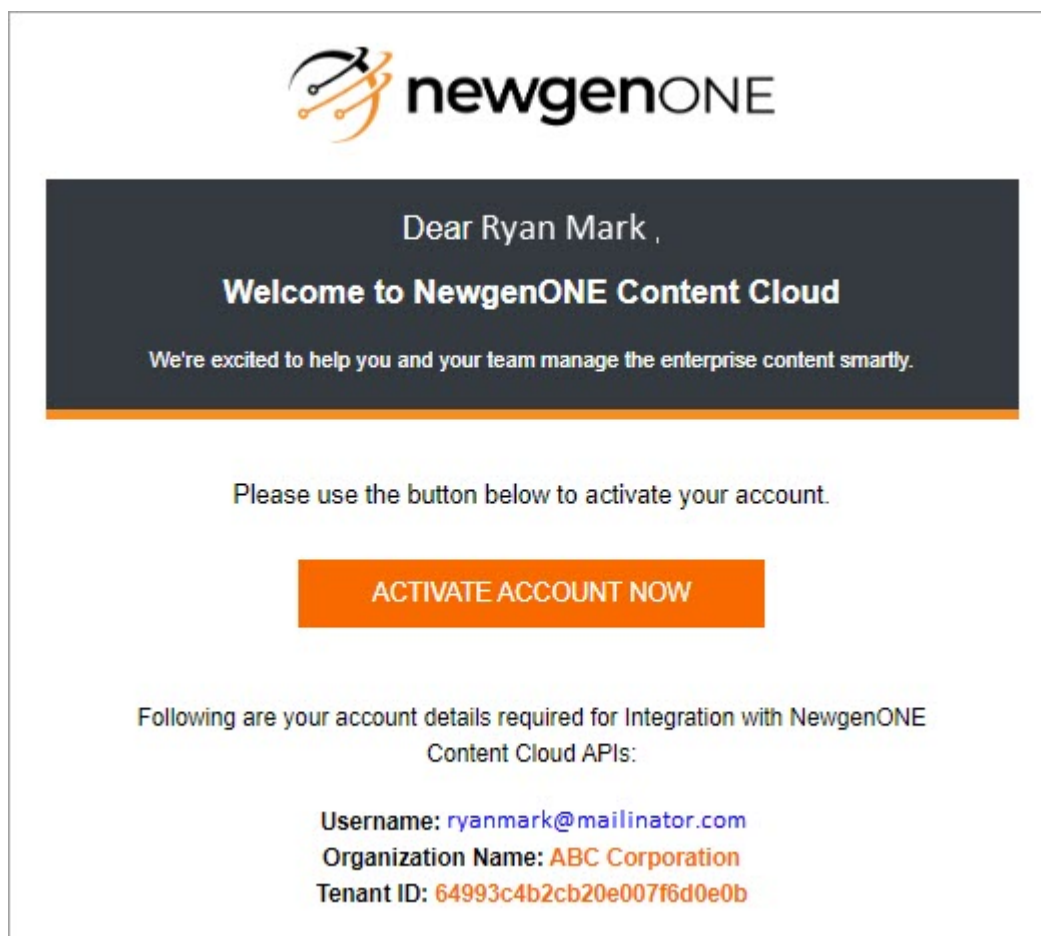
Perform the below steps to register a new tenant account:

1. On the **NewgenONE Content Cloud** web page, click **Try for free**. The page to sign up for a tenant account appears.



2. Fill in the required details. Fields marked with * are mandatory to fill. The Password must follow the below criteria:
 - At least 1 capital letter.
 - At least 1 numeric.
 - At least 1 punctuation.
 - At least 8-16 characters.

- No space allowed.
3. Click **Sign Up** to receive an activation link on your registered email address.
 4. Go to your registered email and open the activation link mail.




5. Click the **ACTIVATE ACCOUNT NOW** link to activate the registered account. Once the account is activated, you can sign in using your registered account details.

Registering an application

Application registration allows you to leverage the OAuth 2.0 authentication and authorization mechanism of NewgenONE Content Cloud for the integration purpose of any custom application. It helps to integrate an application scope rather than a user

scope. This is useful for integrations where users and corresponding rights may be managed by invoking the application.

The Application Registration tab allows you to register an application.

 Only Admin user can register, edit, and delete an application.


Perform the below steps to register an application:

1. On the Application Registration tab, click **+ Register Application**. If there is no application registered earlier, then you can also register an application using the **+ New App Registration** displayed on the right pane. The Register Application page appears. This page comprises four tabs to register an application:
 - Configure Platform
 - Client Credentials
 - Security Settings
 - Summary

By default, the **Configure Platform** tab appears to configure the application.


2. Specify the following fields:
 - **Application Name** — Enter the application name. It must follow the below criteria:
 - It contains a maximum of 1-50 characters
 - Leading or trailing spaces are not allowed.
 - Special characters are not allowed “-” and “_”.
 - Numbers are allowed.
 - **Please Select platform** — Select the required platform where you want to host the registered application. The following options are available:
 - **Web** — Select this option to configure for browser client applications and then follow the below step:
 - Specify the **Redirect URI** where the registered application redirects. The following are the validation criteria of Redirect URI:
 - It must be unique.
 - It must be unique.
 - The character length must be less than 256.
 - It does not contain wildcard characters.
 - It must start with HTTPS or http://localhost.
 - It must be a valid URL.
 - Only 5 redirect URIs can be added.

- Click **Add URI** to add the specified URI.


 Redirect URI associated with a Micro UI cannot be edited or deleted.

- **Server** — Select this option to configure for web or application server-based applications.

3. Click **Next**. The Client Credentials tab appears. This tab allows you to generate a Client Secret to access the registered application.

 Secret associated with a Micro UI cannot be deleted.

4. Click **+ Generate Client Secret**. The Client Secret dialog appears. The generated secret code appears in the **Your Generated Secret Code** field.

 If you edit the Expiry Period of a saved Client Secret, then the Secret Code updates automatically. Due to this change, the URL of the associated Micro UI or the OAuth 2.0 client is impacted. It is recommended that you must update the latest Micro UI URL or OAuth 2.0 client wherever it is needed.

5. Select an expiry date from the **Select Expiry Period** dropdown list. Or, click the **Create Custom** link to set a required expiry date that is not available in the dropdown list.

- If you have selected the Server option to host your application as described in Step 2, then an additional field **Select Role** appears.
- Select an appropriate option from the **Select Role** dropdown list.

6. Click **Save** to finalize the expiry period for the generated Secret Code. The added Client Secret code appears in the Client Credentials tab.


 You can Preview, Edit, and Delete the generated Client Secret code using the **Actions** icon  against it.

7. Click **Next**. The Security Settings tab appears.



8. Specify the following information:

- If you have selected the Web option to host your application as described in Step 2, then specify the **Whitelisted Domain** to access the registered application and click **Add** to save the specified whitelisted domain. The following are the validation criteria of Whitelisted Domain URI:

- It must be unique.
- The character length must be less than 256.
- It does not contain wildcard characters.
- It must start with HTTPS or http://localhost.
- It must be a valid URL.

 Whitelisted domains associated with a Micro UI cannot be edited or deleted.

- If you have selected the Server option to host your application as described in step 2, then follow the below steps:
 - From the right pane, add a required whitelisted IP address. It must follow the below criteria:

 You can Edit and Delete the added IP address using the **Actions** icon  against the required IP address.



- It contains a maximum of 7-15 characters limit.
- IP addresses with wildcards are allowed.

Example:

192.12.12.1, 192.12.11.*, 192.12.*, and so on are valid.

192.*, 192.12.12*, and so on are not valid.

- Click **Browse File** that appears on the left pane to upload a certificate from your system. The following certificate file types are supported:
 - crt
 - .cer

 Once the domain is added, you can Edit and Delete the added domain using the **Actions** icon  against it.

9. Click **Finish**. The Summary tab appears. This tab displays the details of the registered application.

Microservices

This section includes:

- [App services](#)
- [User services](#)
- [Role services](#)
- [Folder services](#)
- [Content services](#)
- [Secret key services](#)
- [Security classification services](#)
- [Data class services](#)
- [Notes services](#)
- [Search service](#)
- [Annotation services](#)
- [Document Services](#)
- [Sample code](#)
- [Common error codes](#)

App services

App services allow you to leverage the OAuth 2.0 authentication and authorization mechanism of NewgenONE Content Cloud for the integration purpose of any custom application. It helps to integrate an application scope rather than a user scope. This is useful for integrations where users and corresponding rights may be managed by invoking the application.

Prerequisites:

- A tenant ID is required to access the NewgenONE Content Cloud services. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
- Parameter org is required, its value is *ECM* to try out any API.

- A system (Type: SYS) app is created while registering for a new tenant. All NewgenONE Content Cloud portal users are authenticated through this app. The user can also create other apps for integration purposes.

App service contains the following API endpoints for authentication:

- [Code API](#)
- [Token API](#)
- [Getting an app by ID](#)
- [Getting app list](#)
- [Generate auth code](#)
- [Generate access token](#)
- [Validate access token](#)
- [Delete access token](#)

These APIs can be used to create *accessToken* and *refreshToken*.

Headers

The below table lists the header parameters for app services:

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
userId	Conditional		String	<p>This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed.</p> <p>Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.</p>
appId	Mandatory		String	<p>This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.</p>

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Code API

The *Code* API allows you to obtain the authorization code that helps you to obtain an authentication token.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/auth/code*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Headers

```
{
  "Accept": "application/json",
  "org": "ECM",
  "Content-Type": "application/json",
  "appId": "625e99f98709d93b3d65f925",
  "tenantId": "625e99ae6558276313b289f7"
}
```

The following are descriptions of header parameters:

Parameter	Description
Accept	It indicates the format of the request body and response body. For example, application/json.
org	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.
Content-Type	It indicates the type of content that is acceptable to pass in a sample request.
appld	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.

Sample request

```
{
  "emailId": "sampleEmail@emailserver.com",
  "password": "base64encodedpasswordXXXX",
  "redirect_uri": "https://ecmnext.azurewebsites.net/"
}
```

The following table describes the request parameters of *Code* API:

Parameter	Mandatory/ Optional	Description	Data Type
emailId	Mandatory	Email ID of the user.	String
password	Mandatory	The password of the above user.	String

Parameter	Mandatory/ Optional	Description	Data Type
redirect_uri	Mandatory	It is the redirect domain URL where the request will be redirected after authentication.	String

Sample response

```
{
  "redirect_uri": "https://ecmnext.azurewebsites.net/?code=42fe7a2a-6f63-4886-
b6cb-e5367ec99d43"
}
```

The following table describes the request parameters of *Code* API:

Parameter	Description
redirect_uri	It is the redirect domain URL where the request will be redirected after authentication. It provides the domain's URL along with code which is further used in the request body in the token API.

Token API

The *Token* API allows you to obtain an access token.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/auth/token*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Headers

```
{
  "Accept": "application/json",
  "org": "ECM",
  "Content-Type": "application/json",
  "appId": "625e99f98709d93b3d65f925",
  "secret": "3yAldMKEi5pNfAK4TRLEXmAltoRNkI31pj0QPINDIGJWSx3uaGjT10GgNIM6Zp3T",
  "tenantId": "625e99ae6558276313b289f7",
  "type": "CODE"
}
```

The following are descriptions of header parameters:

Parameter	Description
Accept	It indicates the format of the request body and response body. For example, application/json
org	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.
Content-Type	It indicates the type of content that is acceptable to pass in a sample request.
appId	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	The secret is required for security purposes. An app is secured by creating a secret with an expiry. Furthermore, an app can contain multiple secrets with different expiry.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
type	This parameter specifies the type to create an access token. For example: type=code (Create accessToken from code for Web to Server flow.) type=jwt (Create accessToken from JWT token for Server to Server flow.) type=refreshToken (Create accessToken from refreshToken for Web to Server flow.)

Sample request

For creating accessToken from code:

```
{
  "code": "42fe7a2a-6f63-4886-b6cb-e5367ec99d43"
}
```

For creating accessToken from JWT token:

```
{
  "jwt": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJkaW1hbnNodSI6ImNoZWNrQGlhaWxkcm9wLmNjIiwiaWF0IjoxNjUyODc3fQ.MX6aj9oaaCG1AWYaqVatGePJZ6HM7Heka9gBETBPeTuvNdVLTkEvGoluhxELKjNqyXdAdlzYyucyZ93BuZzXizknMEffq5EvV58eWyJgGoecSbH-bpRZXx4mhLbaQaDbRefaoewYhFzyX8jop4FS6pkLgrylhwmfTZUo-Kp8QTE_I-9B_JaqvxSFxJXXYqYmIHFwOpEWKjUiL7eh0uwCfaQq_TwYmyPxEO7m9QaKdDnx1E8NxZNYOJRHNRHmtSkG7Dekd0xnMJwFgztlSODAhnx2hCT6zCMixvbwqb4unPRCiky478bSdFXnZGjIxq8aPFmaO6bNxmKpVmk53A"
}
```

For creating accessToken from refreshToken:

```
{
  "refreshToken": "0.AT8AU1DBECX4Th.....elw9oCuf7azeQgTYea"
}
```

The following table describes the request parameter of *Token* API:

Parameter	Mandatory/Optional	Description	Data Type
code	Mandatory	It is the code used to access the token API.	String

Sample response

```
{
  "accessToken": "eyJ0eXAiOiJKV1QiL.....3rlTgEeEsvVI5Bg",
  "refreshToken": "0.AT8AU1DBECX4Th.....elw9oCuf7azeQgTYea",
  "userId": "6037903d0402fa7ff31d60f8",
  "userEmail": "sampleEmail@emailserver.com",
  "displayName": "sample",
  "tenantId": "60379039920c82760cbf9bee",
  "tenantStorageCredentialId": "603f2c6e3a02a00001b2c375",
  "repositoryId": "603f2c6fab5fc60dad3962bb",
  "tenantName": "sampleEmail",
  "tenantTrashId": "603f2c6fab5fc60dad3962bc",
  "tenantTrialDaysRemaining": 10
}
```

The following table describes the response parameters of *Token* API:

Parameter	Description
accessToken	The accessToken is an authentication token.
refreshToken	The refreshToken is an authentication token.
userId	The userId of the user who initiates the request.
userEmail	User's email ID.
displayName	Name of the user.
tenantId	ID of the tenant.
tenantStorageCredentialId	ID of the tenant storage credentials.
repositoryId	ID of the repository to store the content.
tenantName	Name of the tenant.
tenantTrialDaysRemaining	It displays the number of days remaining for your free trial on NewgenONE Content Cloud platform.

Getting app by ID

The *Get app by ID* API allows you to fetch an app using its app ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/app/{{id}}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling *Get app by ID* API, pass the *id* parameter in the request URL. For example, if you want to fetch an app whose ID is 62bc4ef9e7c42f246f4ca46a, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/appservice/app/
62bc4ef9e7c42f246f4ca46a
```

The following table lists the descriptions of request parameters of the *Get app by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	string	It indicates the app ID. It is a mandatory parameter to get a single app.	URL

Sample response

```
{
  "id": "62bc4ef9e7c42f246f4ca46a",
  "name": "Customer KYC",
  "type": "SSF",
  "comment": "Application created for customer verification",
  "linkedApplicationId": "itr658",
  "secrets": [
    "igiyhj"
  ],
  "whiteListIPs": [
    {
      "ip": "225.1.4",
      "restriction": false
    }
  ],
  "whiteListDomains": [
    "https://ecmnext.azure.net/"
  ],
  "redirectURLs": [
    "https://ecmnext.azure.net/"
  ],
  "tenantId": "6285ecledda86d3e391775fc",
  "ownerId": "6285ec202c8d053ffa72464f",
  "creationDateTime": "2022-06-29T13:09:13.136Z",
  "revisedDateTime": "2022-06-29T13:09:13.136Z",
  "accessDateTime": "2022-07-19T13:09:13.136Z"
}
```

The following table lists the descriptions of response parameters of the *Get app by ID* API:

Parameter	Description
id	ID of the created app.
name	Name of the app. The first character of the app name must be an alphabet.
type	It indicates the type of the app flow.
comment	It provides a brief description for creating the application.
linkedApplicationId	It stores and links the micro UI and its ID to the created app. The value of this parameter only allows alphanumeric characters. It can be empty.

Parameter	Description
secrets	It provides a list of the secret codes. It can be null or empty. It accepts alphanumeric, plus, equal, and forward slash.
whiteListIPs	It provides a list of whitelisted IPs. It can be empty and null.
whiteListDomains	It provides a list of whitelisted domains. It can be empty and null.
redirectURLs	It provides a list of redirect URLs where the request redirects after authentication.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the app.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the app was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the app was last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the app was last accessed.

Getting app list

The *Get app list* API allows you to fetch the list of all apps using its tenant ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/app*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get app list* API, set the value of the *sortOn* and *sortOrder* parameters in the request params.

The following table lists the descriptions of request parameters of the *Get app list* API:

Parameter	Mandatory/Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
sortOn	Optional	All possible fields			String	It indicates the field name based on which the response will be sorted.	Request Param
sortOrder	Optional	ASC - Ascending, DESC - Descending			String	Order of sorting.	Request Param

Sample response

```
{
  [
    {
      "id": "62bc4ef9e7c42f246f4ca46a",
      "name": "Customer KYC",
      "type": "SSF",
      "comment": "App created for customer verification",
      "linkedApplicationId": "itr658",
      "secrets": [
        "igiyhj"
      ],
      "whiteListIPs": [
        {
          "ip": "225.1.4",
          "restriction": false
        }
      ],
      "whiteListDomains": [
        "https://ecmnext.azure.net/"
      ],
      "redirectURLs": [
        "https://ecmnext.azure.net/"
      ],
    }
  ]
}
```

```

"tenantId": "6285ec1edda86d3e391775fc",
"ownerId": "6285ec202c8d053ffa72464f",
"creationDateTime": "2022-06-29T13:09:13.136Z",
"revisedDateTime": "2022-06-29T13:09:13.136Z",
"accessDateTime": "2022-06-29T13:09:13.136Z"
},
{
  "id": "6325a2505681703cf1e2afd9",
"name": "LoginApp",
"type": "SYS",
"comment": "Default APP for user login",
"secrets": [
  "TElPbDFXYlFueTYyeTY2WVRhTVVPRnFqMnViSWRabzBUOVZySXZqUXBCcjF5WEVMSl
  NsdFARszJTVDYxOTVfVg"
],
"whiteListDomains": [
  "https://ecmnext-qa.azurewebsites.net/tenantreport/login",
  "https://ecmnext-qa.azurewebsites.net/documentation/login",
  "https://ecmnext-qa.azurewebsites.net/sampleapp/login",
  "https://ecmnext-qa.azurewebsites.net/microUI/login",
  "https://ecmnext-qa.azurewebsites.net/"
],
"redirectURLs": [
  "https://ecmnext-qa.azurewebsites.net/tenantreport/login",
  "https://ecmnext-qa.azurewebsites.net/documentation/login",
  "https://ecmnext-qa.azurewebsites.net/sampleapp/login",
  "https://ecmnext-qa.azurewebsites.net/microUI/login",
  "https://ecmnext-qa.azurewebsites.net/"
],
"tenantId": "6325a1e3d33f86469bd35844",
"ownerId": "System",
"creationDateTime": "2022-09-17T10:32:48.849Z",
"revisedDateTime": "2022-09-17T10:32:48.849Z",
"accessDateTime": "2022-09-17T10:32:48.849Z"
}
]
}

```

The following table lists the descriptions of response parameters of the *Get app list* API:

Parameter	Description
id	ID of the created app.
name	Name of the app. The first character of the app name must be an alphabet.
type	It indicates the type of the app flow.
comment	It provides a brief description for creating the application.

Parameter	Description
linkedApplicationId	It stores and links the micro UI and its ID to the created app. The value of this parameter only allows alphanumeric characters. It can be empty.
secrets	It provides a list of the secret codes. It can be null or empty. It accepts alphanumeric, plus, equal, and forward slash.
whiteListIPs	It provides a list of whitelisted IPs. It can be empty and null.
whiteListDomains	It provides a list of whitelisted domains. It can be empty and null.
redirectURLs	It provides a list of redirect URLs where the request redirects after authentication.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the app.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the app was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the app was last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the app was last accessed.

Generating auth code

The *Generate auth code API* allows you to generate an authentication code for a registered app.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/auth/code*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "emailId": "james@mailinator.com",
  "redirect_uri": "https://ncc-qa.azurewebsites.net/",
  "password": "amFtZXMxOTc0Iw=="
}
```

The following table lists the descriptions of request parameters of the *Generate auth code API*:

Parameter	Mandatory/Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
emailId	Mandatory				String	Email ID of the user.	Request body
password	Mandatory				String	Password of the user in base64 encoded format.	Request body
redirect_uri	Mandatory				String	Redirect URL that is provided while app registration.	Request body

Parameter	Mandatory/ Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
appId	Mandatory				String	App ID through which the code is to be generated.	Header
tenantId	Mandatory				String	Tenant ID is a system-generated ID, it generates when a client is registered at the NewgenONE Content Cloud Platform. This parameter is expected in the header of every API call.	Header

Sample response

```
{
  "redirect_uri": "https://ncc-qa.azurewebsites.net/?code=333dd388-36fa-4676-b9c6-a8c7e8928452"
}
```

The following table lists the description of the response parameter of the *Generate auth code* API:

Parameter	Description
redirect_uri	This parameter returns the redirect URL sent in the request body along with generated code.

Generating access token

The *Generate access token* API allows you to generate an access token. It generates a token based on code, refresh token, or JWT as auth flow.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/auth/token*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

For code flow:

```
{
  "code": "333dd388-36fa-4676-b9c6-a8c7e8928452"
}
```

For refresh token flow:

```
{
  "refreshToken": "{{refreshTokenGlobal}}"
}
```

The following table lists the descriptions of request parameters of the *Generate access token* API:

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
appId	Mandatory				String	The app ID through which the code is to be generated.	Header
tenantId	Mandatory				String	Tenant ID is a system-generated ID, it generates when a client is registered at the NewgenONE Content Cloud Platform. This parameter is expected in the header of every API call.	Header
secret	Mandatory				String	Secret key associated with the app through which the accessToken is to be created.	Header

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
type	Mandatory				String	<p>This parameter specifies the type to create an access token. For example:</p> <p>type=code (Create accessToken from code for Web to Server flow.)</p> <p>type=jwt (Create accessToken from JWT token for Server to Server flow.)</p> <p>type=refreshToken (Create accessToken from refreshToken for Web to Server flow.)</p>	Header
code	Conditional				String	<p>Code obtained using Code API.</p> <p>Conditions:</p> <p>type=code (Create accessToken from code for Web to Server flow.)</p> <p>type=jwt (Create accessToken from JWT token for Server to Server flow.)</p> <p>type=refreshToken (Create accessToken from refreshToken for Web to Server flow.)</p>	Request body

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
jwt	Conditional				String	<p>The JWT is created using the certificate and private key.</p> <p>Condition:</p> <p>type=code (Create accessToken from code for Web to Server flow.)</p> <p>type=jwt (Create accessToken from JWT token for Server to Server flow.)</p> <p>type=refreshToken (Create accessToken from refreshToken for Web to Server flow.)</p> <p>Steps to obtain accessToken in Server to Server flow</p> <ol style="list-style-type: none"> 1. Create an APP. 2. Create secret and associate roles to it. 3. Create a self-signed .crt certificate. The steps to generate a certificate: <ol style="list-style-type: none"> a. Create private key -openssl genrsa -out domain.key 2048. 	Request body

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
						<ul style="list-style-type: none"> b. Create certificate signing request -openssl req -key domain.key -new-out domain.csr c. Create certificate -opencertificate-signkey domain.key -in domain.csr -req -days 365 -out domain.crt. d. Save the private key in the domain.key securely which will be used for creating JWTUpload the certificate in the APP. <ol style="list-style-type: none"> 4. Use a private key, JWT.io, or any other means to create a JWT token. 5. The algorithm used is RSA256 6. The JWT must contain emailId = tenant Owner email ID in the payload. 7. Use all these parameters along with JWT for obtaining accessToken. 	

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
refreshToken	Conditional				String	<p>Refresh Token obtained while generating accessToken using Code API.</p> <p>Condition: type=code (Create accessToken from code for Web to Server flow.) type=jwt (Create accessToken from JWT token for Server to Server flow.) type=refreshToken (Create accessToken from refreshToken for Web to Server flow.)</p>	Request body

Sample response

The following table lists the description of the response parameter of the *Generate access token* API:

Parameter	Description
accessToken	The accessToken is an authentication token.
refreshToken	The refreshToken is an authentication token.
userId	The userId of the user who initiates the request.
userEmail	User's email ID.
displayName	Name of the user.
tenantId	ID of the tenant.

Parameter	Description
tenantStorageCredentialId	ID of the tenant storage credentials.
repositoryId	ID of the repository to store the content.
tenantName	Name of the tenant.
tenantTrashId	Trash ID of the tenant.
globalAdmin	Indicates whether the user is a global admin or not.
tenantTrialDaysRemaining	It displays the number of days remaining for a free trial on the NewgenONE Content Cloud platform.
welcomeEnabled	Indicates whether the welcome message is enabled or not,
isAdmin	Indicates whether the user is of an admin role or not.
expiryDateTime	Indicates the expiry date and time of the generated token.
orgName	Name of the organization.

Validating access token

The *Validate access token API* allows you to validate an access token.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/auth/token/validate*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Validate access token API*, pass the value of the *accessToken* parameter in the header to validate the required access token.

The following table lists the descriptions of request parameters of the *Validate access token* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
accessToken	Mandatory				String	The accessToken is an authentication token. It is obtained while signing in for the NewgenONE Content Cloud platform.	Header

Sample response

```
{
  "expiryDateTime": "2022-10-31T21:59:22.116Z",
  "userId": "633eb61338c79b62379647b2",
  "securityClearance": {
    "SC_Secret-GT_Task": {
      "read": true,
      "write": true,
      "modify": true,
      "delete": true,
      "secured": true
    }
  },
  "isAdmin": true
}
```

The following table lists the descriptions of response parameters of the *Validate access token* API:

Parameter	Description
expiryDateTime	Indicates the expiry date and time of the generated access token.

Parameter	Description
userId	ID of the user.
securityClearance	This parameter checks the security clearance for the assigned rights to a specific role.
isAdmin	Indicates whether the user is of an admin role or not.

Deleting access token

The *Delete access token API* allows you to delete an access token.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/appservice/auth/token*
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete access token API*, pass the value of the *accessToken* parameter in the header to delete a specific access token.

The following table lists the descriptions of request parameters of the *Delete access token API*:

Parameter	Mandatory/Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
accessToken	Mandatory				String	The accessToken is an authentication token. It is obtained while signing in for the NewgenONE Content Cloud platform.	Header

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Error codes

The below table lists the error codes for app services:

Error Code	Error message
28001	App creation error.
28002	App does not found.
28004	Session error.
28005	Invalid redirect URI.
28006	Invalid secret.
28007	Invalid code.
28008	Invalid app.
28009	Invalid JWT token.
280010	Invalid access token.
280011	Invalid refresh token.
280012	Invalid role ID.
280013	Token expired.
280014	Invalid Auth flow type.
280015	Certificate does not found.
280016	Certificate parsing.
280017	Invalid login credentials.
280018	System APP can't be updated.
280019	System APP can't be deleted.

Error Code	Error message
280021	App type SYS is not allowed.
280022	Invalid redirect URL.
280023	Secret does not exists.
280024	Secrets field is not allowed in this API call.
280025	Duplicate App creation is not allowed.
280026	App name can not be null or empty.
280027	Invalid App ID.
280028	Redirect URL required for WSF type.
280029	Invalid App type.
280030	Secret must not be blank or null.
280031	Invalid sort on value.
280032	Invalid sort order.
280033	Invalid certificate extension.
280034	Code is missing.
280035	Refresh token is missing.
280036	Password is not in Base64 encoded format
280037	File size must not exceed 100 KB.

User services

User service allows you to manage user operations. It deals with operations like user creation, updating, sync, deletion, and so on.

Prerequisites:

- A tenant ID is required to access the NewgenONE Content Cloud services. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
- Parameter org is required, its value is ECM to try out any API.

The User services include the following APIs:

- [Get user list](#)
- [Get user details](#)

Headers

The below table lists the header parameters for user services:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
tenantId	Mandatory				String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform.	Header
Content-Type	Mandatory	application/json			String	Type of the content.	Header
accessToken	Mandatory				String	AccessToken is an authentication token. It is obtained from the login API.	Header

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
org	Mandatory	ECM			String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.	Header
userId	Conditional				String	This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a	Header

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
						result, the request does not fail if the <code>userId</code> parameter is not passed. Since there is no way to access the <code>userId</code> parameter from an <code>accessToken</code> in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.	
<code>appId</code>	Conditional				String	This parameter defines the App ID of the application through which the <code>accessToken</code> is created using Auth or Token API.	Header

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
secret	Conditional				String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.	Header

Getting users list

The *Getting users list* API allows you to get a list of all the users under a tenant.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/userregisterservice/users*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Getting users list* API, pass the value of the *search* parameters in the request URL. For example, if you want to get the list of users in a folder whose ID is *630e6e6cbe4bd96a2a580b65*, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/userregisterservice/users?
folderId=630e6e6cbe4bd96a2a580b65
```

The following table describes the request parameter description of the *Getting users list* API:

Parameter	Mandatory/Optional	Description	Data Type	Parameter Type
Search Parameters	Optional	The search parameters to find the user details. For example, parentFolderId, folderId, and so on.	String	Query parameter

Sample response

```
[
  {
    "id": "62d63eb7016b20476a1c6ae4",
    "displayName": "Bhavya",
    "tenantDetails": [
      {
        "tenantId": "62d63eb5a2cf20503b23c2ce",
        "tenantName": "W3 Markting Ltd."
      }
    ],
    "emailId": "bhavya@mailinator.com",
    "isActive": true,
    "creationDateTime": "2022-08-19T05:18:47.251Z",
    "revisedDateTime": "2022-08-19T05:19:10.111Z",
    "accessDateTime": "2022-08-19T05:19:10.111Z",
    "firstTimePasswordReset": true,
    "authType": "AD",
    "welcomeEnabled": true
  },
  {
    "id": "630e6d8fbe4bd96a2a580b63",
    "displayName": "John",
    "tenantDetails": [
      {
        "tenantId": "62d63eb5a2cf20503b23c2ce",
        "tenantName": "W3 Markting Ltd."
      }
    ]
  }
],
```

```

    "emailId": "john@mailinator.com",
    "isActive": true,
    "creationDateTime": "2022-08-30T20:05:35.467Z",
    "isUserInvited": true,
    "firstTimePasswordReset": false,
    "authType": "L",
    "welcomeEnabled": true
  },
  {
    "id": "630e6e6cbe4bd96a2a580b65",
    "displayName": "James",
    "tenantDetails": [
      {
        "tenantId": "62d63eb5a2cf20503b23c2ce",
        "tenantName": "W3 Markting Ltd."
      }
    ],
    "emailId": "james@mailinator.com",
    "isActive": true,
    "creationDateTime": "2022-08-30T20:09:16.745Z",
    "isUserInvited": true,
    "firstTimePasswordReset": false,
    "authType": "L",
    "welcomeEnabled": true
  }
]

```

The following table describes the response parameters description of the *Getting users list* API:

Parameter	Description
id	The ID of the user.
displayName	Indicates the user name.
tenantDetails	Indicates the user's tenant details.
emailId	Email ID of the user.
isActive	Indicates the user activation is enabled or disabled.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the user was created.
isUserInvited	It indicates the user invitation is enabled or disabled.
firstTimePasswordReset	It indicates the first-time password reset by the invited user is enabled or disabled.
authType	It indicates the type of user authentication.

Parameter	Description
welcomeEnabled	It provides the enable or disable status of the welcome message.

Getting user details

The *Getting user details* API allows you to get your user details.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/userregisterservice/users/id*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Getting user details* API, pass the value of the *emailId* parameter in the request URL. For example, if you want to get the details of a user whose ID is *630e6e6cbe4bd96a2a580b65*, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/userregisterservice/users/630e6e6cbe4bd96a2a580b65

The following table describes the request parameter description of the *Getting user details* API:

Parameter	Mandatory/Optional	Description	Data Type	Parameter Type
id	Mandatory	The ID of the user to fetch the user details.	String	URL

Sample response

```
[
  {
    "id": "630e6e6cbe4bd96a2a580b65",
    "displayName": "Amit",
    "tenantDetails": [
      {
        "tenantId": "62d63eb5a2cf20503b23c2ce",
        "tenantName": "W3 Marketing Ltd."
      }
    ],
    "emailId": "amit@sharklazer.com",
    "isActive": true,
    "creationDateTime": "2022-08-30T20:09:16.745Z",
    "isUserInvited": true,
    "firstTimePasswordReset": false,
    "authType": "L",
    "welcomeEnabled": true
  }
]
```

The following table describes the response parameters description of the *Getting user details* API:

Parameter	Description
id	The ID of the user.
displayName	Indicates the user name.
tenantDetails	Indicates the user's tenant details.
emailId	Email ID of the user.
isActive	Indicates the user activation is enabled or disabled.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the user was created.
isUserInvited	It indicates the user invitation is enabled or disabled.
firstTimePasswordReset	It indicates the first-time password reset by the invited user is enabled or disabled.
authType	It indicates the type of user authentication.

Parameter	Description
welcomeEnabled	It provides the enable or disable status of the welcome message.

Error codes

The following table describes the error codes and their description related to user services:

Error Code	Message
31001	User does not exist.
31002	User account is not active.
31003	User account is already active.
31004	Invalid email ID or password.
31005	Password expired.
31006	Authentication failed.
31007	Error creating Cabinet.
31008	Error creating Credentials.
31009	Error creating Asset folder.
310010	Error creating Trash folder.
310011	Error updating Tenant.
310012	Error sending Email.
310013	Invalid Password Token.
310014	Access Token is missing.
310015	Invalid Access Token.
310016	Invalid Refresh Token.
310017	Error updating Password.
310018	AuthType is missing.
310019	Redis connection exception.

Error Code	Message
310020	User is not registered with Google account, try AD method
310021	User registered with Google Sign Up.
310022	Trial period ended for the user.
310023	User Access Token cannot be refreshed.
310024	User cannot be deleted using the provided token.
310025	System user cannot be deleted.
310026	User deletion failed.
310027	Failed to delete tenant of the user.
310028	Invalid Request
310029	TenantID cannot be blank.
310030	Activation token is not valid
310031	Admin user cannot be deleted.
310032	User already exists.
310033	Unable to register user to AD.
310034	User must reset password. Temporary password can be used only once.
310035	Unable to proceed. User does not have necessary authorization.
310036	Error processing request.
310037	Unable to create Group.
310038	Unable to update Group details.
310039	Multiple groups exists with this name.
310040	No Organisations found for this user.
310041	Required user to change password on first login.
310042	Password cannot be updated for this user.
310043	Invalid OTP.
310044	Email ID not registered.

Error Code	Message
310045	OTP expired.

Role services

Roles services are used to create and manage the type of access a user has in an organization.

For example, a user may have access across all folders and files, while another user may have only read permission, and other users may have both read and write permissions. It is to be used in conjunction with a security classification. You can associate the created role with an application or a user.

The role services include the following APIs:

- [Get role by ID](#)
- [Get all roles](#)
- [Get role for multiple user IDs](#)
- [Get multiple roles by role IDs](#)

Headers

The below table lists the header parameters for role services:

Parameter	Mandatory/ Optional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/ json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.

Parameter	Mandatory/ Optional	Accepted Values	Data Type	Description
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameter	Mandatory/ Optional	Accepted Values	Data Type	Description
userId	Conditional		String	<p>This parameter defines the user ID of the logged-in user.</p> <p>Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed.</p> <p>Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent.</p> <p>Otherwise, services may raise an exception.</p>

Parameter	Mandatory/Optional	Accepted Values	Data Type	Description
appId	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Getting role by ID

The *Get role by ID* API allows you to delete an existing role from the system using the role ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/rolesservice/role/{{id}}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get role by ID* API, pass the value of the *id* parameter in the request URL. For example, if you want to get a role whose ID is `63259803751fcc04dc20dbe0`, then pass the request URL as follows:

`https://<IP address>:<Port>/ecmapi/rolesservice/role/63259803751fcc04dc20dbe0`

The following is the description of the request parameter of the *Get role by ID* API:

Parameter	Mandatory/Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	It indicates the role ID which you want to get. It accepts only alphanumeric values.	URL

Sample response

```
{
  {
    "id" : "63259803751fcc04dc20dbe0",
    "tenantId" : "6325977c59f6e45f94a890f2",
    "ownerId" : "6325977f737e33746b041fd4",
    "name" : "IT Admin",
    "comment" : "Role to perform admin level activities",
    "userIds" : [
      "63259985737e33746b041fd7",
      "63259fca737e33746b041fdb",
      "6325a191737e33746b041fdd"
    ],
    "securityClearance" : {
      "GT_Officer-SC_Top Secret" : {
        "read" : true,
        "write" : true,
        "modify" : true,
        "delete" : true,
        "secured" : true
      }
    },
    "creationDateTime" : "2022-09-17T09:48:51.323Z",
    "revisedDateTime" : "2022-09-17T10:29:37.983Z",
    "accessDateTime" : "2022-09-22T10:29:37.983Z"
  }
}
```

The following table lists the descriptions of response parameters of the *Get role by ID* API:

Parameter	Description
id	ID of the created role.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the role.
name	Name of the role.
comment	It provides a brief description for creating the application.
userIds	List of user IDs for the created role.
securityClearance	This parameter checks the security clearance for the assigned rights to a specific role.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the role was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the role was the last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the role was last accessed.

Getting all roles

The *Get all roles* API allows you to get all existing roles in a system using tenant ID.

- **Service URL:** *https://<IP address>:<Port>/ecmapi/rolesservice/role*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get all roles* API, pass the value of the `tenantId` parameter in the request URL. For example, if you want to get the list of all existing roles in a tenant whose ID is `6037903d0402fa7ff31d60f8`, then pass the request URL as follows:

```
https://<IP address>:<Port>/ecmapi/rolesservice/role?
sortOn=name&sortOrder=ASC&userIdForRoles=6037903d0402fa7ff31d60f8
```

The following is the description of the request parameter of the *Get all roles* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
sortOn	Optional		creationDateTime		String	Secret key is to be sorted on basis of the value of sortOn field. It accepts all defied fields as the value.	Request param
sort Order	Optional	ASC - Ascending DESC - Descending	DESC		String	It defines the sorting order of roles in the list.	Request param

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
userId ForRoles	Optional				String	<p>If given, to search all the roles that contain the given userId. Or else returns all available roles by tenantId. It accepts only alphanumeric values. If space is present anywhere it will be considered invalid. It is an optional parameter, and needs to be searched only if given in the parameters.</p>	Request param

Sample response

```

{
  [
    {
      "id" : "63259803751fcc04dc20dbe0",
      "tenantId" : "6325977c59f6e45f94a890f2",
      "ownerId" : "6325977f737e33746b041fd4",
      "name" : "IT Admin",
      "comment" : "Role created to perform admin level activities",
      "userIds" : [

```



```

        "63259fca737e33746b041fdb",
        "6325a191737e33746b041fdd"
    ],
    "securityClearance" : {
        "GT_Officer-SC_Top Secret" : {
            "read" : true,
            "write" : true,
            "modify" : true,
            "delete" : false,
            "secured" : true
        }
    },
    "creationDateTime" : "2022-09-17T09:48:51.323Z",
    "revisedDateTime" : "2022-09-27T10:29:37.983Z",
    "accessDateTime" : "2022-09-27T10:29:37.983Z"
},
{
    "id" : "63259803751fcc04dc2dfe0",
    "tenantId" : "6325977c59f6e45f94a890f2",
    "ownerId" : "6325977f737e33746b041fd4",
    "name" : "IT Manager",
    "comment" : "Role created to manage IT issues",
    "userIds" : [
        "63259985737e33746b041fd7",
        "63259fca737e33746b041fdb",
        "6325a191737e33746b041fdd"
    ],
    "securityClearance" : {
        "GT_Officer-SC_Top Secret" : {
            "read" : true,
            "write" : true,
            "modify" : true,
            "delete" : false,
            "secured" : true
        }
    },
    "creationDateTime" : "2022-09-18T09:48:51.323Z",
    "revisedDateTime" : "2022-09-18T10:29:37.983Z",
    "accessDateTime" : "2022-09-18T10:29:37.983Z"
}
]
}

```

The following table lists the descriptions of response parameters of the *Get all roles* API:

Parameter	Description
id	ID of the created role.
tenantId	ID of the registered tenant.

Parameter	Description
ownerId	ID of the user who creates the role.
name	Name of the role.
comment	It provides a brief description for creating the application.
userIds	List of user IDs for the created role.
securityClearance	This parameter checks the security clearance for the assigned rights to a specific role.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the role was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the role was the last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the role was last accessed.

Getting roles for multiple user IDs

The *Get roles for multiple user IDs* API allows you to get all roles that contain the provided `userId(s)`.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/rolesservice/role/users`
- **Method type:** POST
- **Request content type:** `application/json`
- **Response content type:** `application/json`

Sample request

Pass the request URL as follows to get roles for multiple user IDs:

```
https://<IP address>:<Port>/ecmapi/rolesservice/role/users?
sortOn=name&sortOrder=ASC
```

Request body:

```
{
  "userIds" : [
    "6325a191737e33746b041fdd"
  ]
}
```

The following is the description of the request parameter of the *Get roles for multiple user IDs* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	Data Type	Description	Parameter Type
sort On	Optional		creationDateTime	String	Secret key is to be sorted on basis of the value of the sortOn field. It accepts all defied fields as the value.	Request param
sort Order	Optional	ASC - Ascending DESC - Descending	DESC	String	It defines the sorting order of roles in the list.	Request param

Parameter	Mandatory/Optional	Accepted Values	Default Value	Data Type	Description	Parameter Type
userIds	Optional			List<String>	List of user IDs for the provided roles. It accepts only alphanumeric values. If space is present anywhere it will be considered invalid. It is an optional parameter, if provided, it returns roles for all user IDs, or else, it returns all roles by tenant ID.	Request body

Sample response

```
{
  [
    {
      "id" : "63259803751fcc04dc20dbe0",
      "tenantId" : "6325977c59f6e45f94a890f2",
      "ownerId" : "6325977f737e33746b041fd4",
      "name" : "IT Admin",
      "comment" : "Role created to perform admin level activities",
      "userIds" : [
        "63259fca737e33746b041fdb",
        "6325a191737e33746b041fdd"
      ],
      "securityClearance" : {
```

```

        "GT_Officer-SC_Top Secret" : {
            "read" : true,
            "write" : true,
            "modify" : true,
            "delete" : false,
            "secured" : true
        }
    },
    "creationDateTime" : "2022-09-17T09:48:51.323Z",
    "revisedDateTime" : "2022-09-27T10:29:37.983Z",
    "accessDateTime" : "2022-09-27T10:29:37.983Z"
},
{
    "id" : "63259803751fcc04dc2dfe0",
    "tenantId" : "6325977c59f6e45f94a890f2",
    "ownerId" : "6325977f737e33746b041fd4",
    "name" : "IT Manager",
    "comment" : "Role created to manage IT issues",
    "userIds" : [
        "63259985737e33746b041fd7",
        "63259fca737e33746b041fdb",
        "6325a191737e33746b041fdd"
    ],
    "securityClearance" : {
        "GT_Officer-SC_Top Secret" : {
            "read" : true,
            "write" : true,
            "modify" : true,
            "delete" : false,
            "secured" : true
        }
    },
    "creationDateTime" : "2022-09-18T09:48:51.323Z",
    "revisedDateTime" : "2022-09-18T10:29:37.983Z",
    "accessDateTime" : "2022-09-18T10:29:37.983Z"
}
]
}

```

The following table lists the descriptions of response parameters of the *Get roles for multiple user IDs* API:

Parameter	Description
id	ID of the created role.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the role.

Parameter	Description
name	Name of the role.
comment	It provides a brief description for creating the application.
userIds	List of user IDs for the created role.
securityClearance	This parameter checks the security clearance for the assigned rights to a specific role.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the role was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the role was the last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the role was last accessed.

Getting multiple roles by role IDs

The *Get multiple roles by role ID(s)* API allows you to get multiple roles that contain the provided roleId(s).

- **Service URL:** `https://<IP address>:<Port>/ecmapi/rolesservice/role/roles`
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

Pass the request URL as follows to get multiple roles by role IDs:

```
https://<IP address>:<Port>/ecmapi/rolesservice/role/roles?
sortOn=name&sortOrder=ASC
```

Request body:

```
{
  "roleIds" : [
    "63259803751fcc04dc20dbe0",
    "63259803751fcc04dc2dfe0"
  ]
}
```

The following is the description of the request parameter of the *Get multiple roles by role ID(s)* API:

Parameter	Mandatory/ Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
sortOn	Optional		creationDate Time		String	Secret key is to be sorted on basis of the value of the sortOn field. It accepts all defied fields as the value.	Request param
sortOrder	Optional	ASC - Ascending DESC - Descending	DESC		String	It defines the sorting order of roles in the list.	Request param

Parameter	Mandatory/Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
roleIds	Optional				List<String>	List of role IDs for the provided role. It accepts only alphanumeric values. If space is present anywhere it will be considered invalid. It is an optional parameter, it returns roles for all user IDs, or else, it returns all roles by tenant ID.	Request body

Sample response

```
{
  [
    {
      "id" : "63259803751fcc04dc20dbe0",
      "tenantId" : "6325977c59f6e45f94a890f2",
      "ownerId" : "6325977f737e33746b041fd4",
      "name" : "IT Admin",
      "comment" : "Role created to perform admin level activities",
      "userIds" : [
        "63259fca737e33746b041fdb",
        "6325a191737e33746b041fdd"
      ],
      "securityClearance" : {
        "GT_Officer-SC_Top Secret" : {
          "read" : true,
          "write" : true,

```



```

        "modify" : true,
        "delete" : false,
        "secured" : true
    }
},
"creationDateTime" : "2022-09-17T09:48:51.323Z",
"revisedDateTime" : "2022-09-27T10:29:37.983Z",
"accessDateTime" : "2022-09-27T10:29:37.983Z"
},
{
    "id" : "63259803751fcc04dc2dfe0",
    "tenantId" : "6325977c59f6e45f94a890f2",
    "ownerId" : "6325977f737e33746b041fd4",
    "name" : "IT Manager",
    "comment" : "Role created to manage IT issues",
    "userIds" : [
        "63259985737e33746b041fd7",
        "63259fca737e33746b041fdb",
        "6325a191737e33746b041fdd"
    ],
    "securityClearance" : {
        "GT_Officer-SC_Top Secret" : {
            "read" : true,
            "write" : true,
            "modify" : true,
            "delete" : false,
            "secured" : true
        }
    },
    "creationDateTime" : "2022-09-18T09:48:51.323Z",
    "revisedDateTime" : "2022-09-18T10:29:37.983Z",
    "accessDateTime" : "2022-09-18T10:29:37.983Z"
}
]
}

```

The following table lists the descriptions of response parameters of the *Get multiple roles by role ID(s)* API:

Parameter	Description
id	ID of the created role.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the role.
name	Name of the role.
comment	It provides a brief description for creating the application.

Parameter	Description
userIds	List of user IDs for the created role.
securityClearance	This parameter checks the security clearance for the assigned rights to a specific role.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the role was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the role was the last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the role was last accessed.

Error codes

The below table lists the error codes for role services:

Error Code	Error Message
62001	Role creation error.
62002	Role not found.
62003	Invalid Role ID.
62004	Roles not found for tenant ID.
62005	User already exists.
62006	User does not exist.
62007	Access denied.
62008	Duplicate role creation not allowed.
62009	Role name must not be blank.
620010	Invalid role name.
620011	Role ID does not exist.
620012	User ID list must not be empty.
620013	User IDs don't exist.

Error Code	Error Message
620014	Role ID list must not be empty.
620015	Role IDs don't exist.
620016	Sort On value is invalid.
620017	Sort Order is not valid.
620018	Security Clearance must not be empty.
620019	Security Classification array must not be empty.
620020	At least one of the security classifications does not exist in the System.
620021	UserId list must not be empty for Admin Role.
620022	UserId must not be blank.
620023	UserId list must contain OwnerId for Admin Role.

Folder services

The folder is used to store and organize documents in the repository. You can add sub-folders and documents within a folder. For example, client applications can call Folder services to create a folder, search for folders, create bulk folders, copy folders, and so on.

Prerequisites:

- A tenant ID is required to access the NewgenONE Content Cloud services. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
- Parameter org is required, its value is ECM to try out any API.
- A repository needs to be created since folders and documents are stored inside a repository where the repository acts as the root of the folder tree hierarchy. Once you log in to the NewgenONE Content Cloud platform, a default repository is created for you.

The folder services includes the following APIs:

- [Add folder](#)
- [Add multiple folders in bulk](#)
- [Modify folder properties](#)
- [Get list of all folders](#)
- [Get folder root path by ID](#)
- [Get list of all immediate child elements for a parent folder](#)
- [Get folder metadata by ID](#)
- [Get folder data class by ID](#)
- [Get folder by ID](#)
- [Copy folder](#)
- [Check copy folder status](#)
- [Move folder to trash](#)
- [Move folder from trash](#)
- [Move folder](#)
- [Search folder on data class](#)
- [Search folders based on metadata](#)
- [Delete folder containing children](#)
- [Delete folder](#)

- [Delete folder data class by ID](#)
- [Delete a folder metadata by ID](#)

Headers

The below table lists the header parameters for folder services:

Parameter	Description	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameter	Description	Accepted Values	Data Type	Description
userId	Conditional		String	<p>This parameter defines the user ID of the logged-in user.</p> <p>Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed.</p> <p>Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.</p>
appId	Mandatory		String	<p>This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.</p>
secret	Mandatory		String	<p>This parameter defines the secret code through which the accessToken is created using Auth or Token API.</p>

Adding folder

The *Add folder* API allows you to create or upload a folder or sub-folder in the folder repository.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "folderName": "Marketing",
  "folderType": "folder",
  "comments": "Workspace for project A",
  "ownerName": "Manoj",
  "ownerId": "AVlFxAVOoeDBAMtAMzee",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "dataclass": {
    "dataClassName": "dataclass_marketing",
    "dataClassId": "62d7f3c3c0ea8e46159b1040"
  },
  "securityClass": [
    "OWNER"
  ]
}
```

The following are the description of the request and response parameters of the *Add folder* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
comments	Optional				String	Comments for the folder.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folder Name	Conditional				Array of string	The array of folder names. It is a mandatory field to create a folder.	Request body
folder Type	Optional	folder		Yes	String	The system operated field. The folder type is 'folder' when you are creating a general folder.	Request body
parent FolderId	Mandatory			Yes	String	The system generated the ID of a parent or root folder.	Request body
used For	Optional	Alphanumeric characters, space, and hyphen only			String	Describe the purpose of the folder.	Request body
metadata	Conditional				String	A JSON Key value pair of metadata associated with the Folder. Mandatory while using API to add metadata to a folder.	Request body

Parameter	Mandatory/ Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
owner Name	Optional				String	The name of the user who creates the folder. The owner name field accepts alphanumeric, hyphen (-), underscore (_), and space. There is no other validation on this field.	Request body
dataclass	Optional				String	A JSON Key value pair of the data indexing associated with the Folder.	Request body
security Class	Optional	Example: ["OWNER", "Department- topsecret"]		Yes	String array	The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
						<p>This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>	

Sample response

```
{
  "id": "630e65b5dfd49e5ebe551674",
  "folderName": "Marketing",
  "folderType": "folder",
  "comments": "Workspace for project A",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "ownerName": "Manoj",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
```

```

"creationDateTime": "2022-08-30T19:32:05.748Z",
"revisedDateTime": "2022-08-30T19:32:05.748Z",
"usedFor": "general",
"dataclass": {
  "dataClassName": "dataclass_marketing",
  "dataClassId": "62d7f3c3c0ea8e46159b1040"
},
"lastModifiedByUserId": "62d63eb7016b20476a1c6ae4",
"securityClass": [
  "OWNER"
]
}

```

The following are the description of the request and response parameters of the *Add folder* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.

Parameter	Description
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field. OWNER is a system tag, which defines that the owner of the document or folder has full rights. Format: ["OWNER", "<Global Tag>-<Security Class>"]

Adding folders in bulk

The *Add folder in bulk* API allows you to create or upload multiple folders or sub-folder in the folder repository.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/bulk*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "folderName": [
    "a"
  ],
  "folderType": "folder",
  "comments": "My workspace for project A",
  "ownerName": "Manoj",
  "ownerId": "AVlFxAVOoeDBAMtAMzee",
  "parentFolderId": "62d63ec46367977f6d813e3c",
```

```

"dataclass": {
  "dataClassName": "dataclass_190fg33ss124",
  "dataClassId": "62d7f3c3c0ea8e46159b1040"
},
"securityClass": [
  "OWNER"
]
}

```

The following are the description of the request parameters of the *Add folder in bulk* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
comments	Optional				String	Comments for the folder.	Request body
folderName	Conditional				Array of string	The array of folder names. It is a mandatory field to create a folder.	Request body
folderType	Optional	folder		Yes	String	The system operated field. The folder type is 'folder' when you are creating a general folder.	Request body
parent FolderId	Mandatory			Yes	String	The system generated the ID of a parent or root folder.	Request body
usedFor	Optional	Alpha Numeric characters, space, and hyphen only			String	Describe the purpose of the folder.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
metadata	Conditional				String	A JSON Key value pair of metadata associated with the Folder. Mandatory while using API to add metadata to a folder.	Request body
ownerName						The name of the user who creates the folder. The owner name field accepts alphanumeric, hyphen (-), underscore (_), and space. There is no other validation on this field.	Request body
dataclass	Optional				String	A JSON Key value pair of the data indexing associated with the Folder.	Request body
securityClass	Optional	Example: ["OWNER", "Department-topsecret"]		Yes	String array	The securityClass field is an array to capture the tags applied to a folder. A	Request body

Parameter	Mandatory/ Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
						<p>combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field. OWNER is a system tag, which defines that the owner of the document or folder has full rights. Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>	

Sample response

```
[
  {
    "id": "630e6614dfd49e5ebe551675",
    "folderName": "a",
    "folderType": "folder",
    "comments": "My workspace for project A",
    "parentFolderId": "62d63ec46367977f6d813e3c",
    "ownerName": "Manoj",
    "ownerId": "62d63eb7016b20476a1c6ae4",
    "tenantId": "62d63eb5a2cf20503b23c2ce",
    "creationDateTime": "2022-08-30T19:33:40.530Z",
    "revisedDateTime": "2022-08-30T19:33:40.530Z",
    "usedFor": "general",
    "dataclass": {
      "dataClassName": "dataclass_190fg33ss124",
      "dataClassId": "62d7f3c3c0ea8e46159b1040"
    },
    "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4",
    "securityClass": [
      "OWNER"
    ]
  }
]
```

The following are the description of the response parameters of the *Add folder in bulk* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.

Parameter	Description
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Modifying folder properties

The *Modify folder properties* API allows you to modify the folder properties.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folders/{{folderId}}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Modify folder properties* API, pass the value of the *id* parameter in the request URL. For example, if you want to modify the folder property whose ID is 630ddd693cf0cb412311e33c, then pass the request URL as follows:

```
http or https://<IP address><Port>/ecmapi/folderservice/folders/630ddd693cf0cb412311e33c
```

Request body:

```
{
  "folderName": "updated name",
  "securityClass": [
    "OWNER",
    "SC_Top Secret-GT_frigate"
  ],
  "dataclass": {
    "test": 13,
    "test3": "2038-10-14T08:57:44.298Z",
    "dataClassId": "63492a6146b74953995877b6",
    "dataClassName": "pack",
    "hvi": "wsww"
  }
}
```

The following are the description of the request parameters of the *Modify folder properties* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	Valid ID of a folder.	URL

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
<Params>	Mandatory			Yes	String	<p>It is used to have the updated values of the folder. All fields sent as part of input JSON in the upload or add folder request, these parameters are available to update other than system-level fields and the fields that are not considered for update. If provided in modifying the folder request, these fields can be ignored and no additional message needs to be shown. Following are the list of parameters:</p> <ul style="list-style-type: none"> • folderType • creationDateTime • accessDateTime • parentFolderId • parentFolder • ownerName • ownerId 	Request body

Sample response

```
{
  "id": "630ddd693cf0cb412311e33c",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Upload documents related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
}
```

```

"ownerId": "62d63eb7016b20476a1c6ae4",
"tenantId": "62d63eb5a2cf20503b23c2ce",
"contentLocationId": "630ddd693cf0cb412311e33b",
"privilege": "INHERITED",
"creationDateTime": 1661853033916,
"revisedDateTime": 1661858582018,
"accessDateTime": 1661858582018,
"version": 1,
"flag": "COMMITTED",
"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"metadata": {
  "key": "v"
},
"dataclass": {
  "dataClassName": "dataclass_banking",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": true,
"indexed": true,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Modify folder properties* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.

Parameter	Description
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents that matches with applied tags in the securityClass field. OWNER is a system tag, which defines that the owner of the document or folder has full rights. Format: ["OWNER", "<Global Tag>-<Security Class>"]

Getting list of all folders

The *Getting list of all folders* API allows you to get the list of all folders for a particular tenant.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders?<SearchQueryParam>*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Case 1: If the tenant has few folder, then pass the below request URL format:

http or https://<IP address>:<Port>/ecmapi/folderservice/folders?{HttpServletRequest}

Where, *HttpServletRequest* specifies the query parameters, for example, "parentFolderId":"62d63ec46367977f6d813e3c"

Case 2: If the tenant has a large number of folders having 62d63ec46367977f6d813e3c parent folder ID, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/folderservice/folders/62d63ec46367977f6d813e3c

In case 2 set the header parameters as provided in the below table:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
offset	Mandatory			Yes	Integer	This parameter is used to set from where the folders list starts to display.	Request header
limit	Mandatory				Integer	This parameter is used to set the number of folders displayed in the list.	Request header

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
sortOn	Mandatory				String	This parameter is used to set the field to sort folders displayed in the list. You can sort the folders using revisedDateTime and name fields.	Request header
sortOrder	Mandatory	ASC or DESC			String	This parameter is used to set the folders sorted list in ascending and descending order. Where ASC specifies the ascending order and DESC specifies the descending order.	Request header

Sample response

```
[
  {
    "id": "630ddd693cf0cb412311e33c",
    "usedFor": null,
    "name": "Banking",
    "contentType": "pdf",
    "comments": "Document related to banking",
    "parentFolderId": "62d63ec46367977f6d813e3c",
    "securityClass": [
      "OWNER"
    ],
    "ownerName": "John",
    "ownerId": "62d63eb7016b20476a1c6ae4",
    "tenantId": "62d63eb5a2cf20503b23c2ce",
    "contentLocationId": "630ddd693cf0cb412311e33b",
    "creationDateTime": 1661853033916,
    "revisedDateTime": 1661853033945,
    "accessDateTime": 1661853033945,
    "version": 1,
    "previousVersion": null,
    "flag": "COMMITTED",
    "noOfPages": "1",
    "documentType": "png",
    "documentSize": "246",
    "metadata": {
      "key": "v"
    }
  }
]
```

```

},
"dataclass": {
  "dataClassName": "dataclass_Home loan",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"versionComments": null,
"checkedOut": false,
"checkedOutBy": null,
"lastCheckedInBy": null,
"lastCheckedInbyUserName": null,
"previousVersionContentId": null,
"checkedOutTime": null,
"parentHierarchy": null,
"docSize": 246
},
{
  "id": "630ddd693cf0cb412311e33c",
  "usedFor": null,
  "name": "Invoice",
  "contentType": "pdf",
  "comments": "Approval certificate",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "Roy",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661853033945,
  "accessDateTime": 1661853033945,
  "version": 1,
  "previousVersion": null,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
},
"dataclass": {
  "dataClassName": "dataclass_Car loan",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,

```



```

"versionComments": null,
"checkedOut": false,
"checkedOutBy": null,
"lastCheckedInBy": null,
"lastCheckedInbyUserName": null,
"previousVersionContentId": null,
"checkedOutTime": null,
"parentHierarchy": null,
"docSize": 246
}
]

```

The following are the description of the response parameters of the *Getting list of all folders* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.

Parameter	Description
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Getting folder root path by ID

The *Getting folder root path by ID* API allows you to fetch the root path hierarchy of a folder using its folder ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/rootpathhierarchy/{folderId}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Getting folder root path by ID* API, pass the value of the *folderId* parameter in the request URL. For example, if you want to get the root path of a folder whose ID is `62d63ec46367977f6d813e3c`, then pass the request URL as follows:

`http or https://<IP address>:<Port>/ecmapi/folderservice/folders/rootpathhierarchy/62d63ec46367977f6d813e3c`

The following is the description of the request parameter of the *Getting folder root path by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	string	ID of the folder to fetch its root folder hierarchy.	URL

Sample response

```
{
  "folder": [
    {
      "comments": "System Folder - Guide Folder",
      "odId": 0,
      "parentFolderId": "62d63ec46367977f6d813e39",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",
      "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
      "folderName": "Explore Sample Application",
    }
  ]
}
```

```

    "id": "62d63ec46367977f6d813e3c",
    "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
  },
  {
    "comments": "System Folder - Tenant Asset Folder",
    "odId": 0,
    "parentFolderId": "62d63ec328268c6a3ee6ba5b",
    "indexed": false,
    "inTrash": false,
    "privilege": "INHERITED",
    "ownerId": "System",
    "version": 1,
    "folderType": "folder",
    "ownerName": "System",
    "tenantId": "62d63eb5a2cf20503b23c2ce",
    "usedFor": "general",
    "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
    "folderName": "TenantAssetFolder",
    "id": "62d63ec46367977f6d813e39",
    "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
  }
],
"cabinet": {
  "comments": "cabinet created by system",
  "odId": 0,
  "indexed": false,
  "inTrash": false,
  "ownerId": "System",
  "accessDateTime": "Tue Jul 19 05:18:59 GMT 2022",
  "version": 1,
  "folderType": "cabinet",
  "ownerName": "System",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "usedFor": "general",
  "revisedDateTime": "Tue Jul 19 05:18:59 GMT 2022",
  "folderName": "DefaultCabinet",
  "id": "62d63ec328268c6a3ee6ba5b",
  "creationDateTime": "Tue Jul 19 05:18:59 GMT 2022"
},
"content": {
  "checkedOut": false,
  "primaryContentId": "630dce060fa8b04c913f1f3e",
  "metadata": {
    "key": "v"
  },
  "flag": "COMMITTED",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "dataclass": {
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "dataClassName": "dataclass_system",
    "age": "20"
  }
}

```

```

},
"documentType": "png",
"privilege": "INHERITED",
"ownerId": "62d63eb7016b20476a1c6ae4",
"documentSize": "246",
"accessDateTime": "Tue Aug 30 09:32:00 GMT 2022",
"ownerName": "John",
"noOfPages": "1",
"id": "630dce060fa8b04c913f1f3e",
"contentType": "pdf",
"contentLocationId": "630dce050fa8b04c913f1f3d",
"latest": false,
"creationDateTime": "Tue Aug 30 08:44:54 GMT 2022",
"securityClass": [
  "OWNER"
],
"accessCount": 0,
"comments": "Document for managment",
"indexed": false,
"nameLowerCase": "system",
"dataClassText": "dataclass_system 62d7f3c3c0ea8e46159b1040 20 ",
"docSize": 246,
"checkedOutBy": "",
"inTrash": false,
"version": 1,
"tenantId": "62d63eb5a2cf20503b23c2ce",
"name": "System",
"revisedDateTime": "Tue Aug 30 09:32:00 GMT 2022"
}
}

```

The following are the description of the response parameters of the *Getting folder root path by ID* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.

Parameter	Description
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Getting list of all immediate child elements for a parent folder

The *Get list of all immediate child elements for a parent folder* API allows you to fetch all the sub-folder available in the required folder.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders/children/{{folderId}}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get list of all immediate child elements for a parent folder* API, pass the value of the `folderId` parameter in the request URL. For example, if you want to get the list of child elements of a folder whose ID is `62d63ec46367977f6d813e3c`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice/folders/children/62d63ec46367977f6d813e3c
```

The following is the description of the request parameter of the *Get list of all immediate child elements for a parent folder* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	The ID of the parent folder to fetch its child elements.	URL

Sample response

```
{
  "folders": [
    {
      "id": "62d63ec46367977f6d813e3c",
      "folderName": "Banking",
      "folderType": "folder",
      "parentFolderId": "62d63ec46367977f6d813e3c",
      "ownerName": "Manoj",
      "ownerId": "62d63eb7016b20476a1c6ae4",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "creationDateTime": "2022-08-09T09:23:44.218Z",
      "revisedDateTime": "2022-08-09T09:23:44.218Z",
      "usedFor": "_",
      "metadata": {
        "k": "v"
      },
      "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
    },
    {
      "id": "62f2281f7262b83bb747b3da",
      "folderName": "Invoice",
      "folderType": "folder",
      "parentFolderId": "62d63ec46367977f6d813e3c",
      "ownerName": "Manoj",
      "ownerId": "62d63eb7016b20476a1c6ae4",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "creationDateTime": "2022-08-09T09:25:51.700Z",
      "revisedDateTime": "2022-08-09T09:25:51.700Z",
      "usedFor": "_",
      "metadata": {
        "k": "v"
      },
      "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
    },
    {
      "id": "62f22bad7262b83bb747b3dc",
      "folderName": "Marketing",
      "folderType": "folder",
      "parentFolderId": "62d63ec46367977f6d813e3c",
      "ownerName": "Manoj",
      "ownerId": "62d63eb7016b20476a1c6ae4",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "creationDateTime": "2022-08-09T09:41:01.272Z",
      "revisedDateTime": "2022-08-09T09:41:01.272Z",
      "usedFor": "_",
      "metadata": {
        "k": "v"
      }
    }
  ]
}
```



```

    },
    "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
  },
  {
    "id": "62f2bad57262b83bb747b410",
    "folderName": "Planning",
    "folderType": "folder",
    "parentFolderId": "62d63ec46367977f6d813e3c",
    "ownerName": "manoj",
    "ownerId": "62d63eb7016b20476a1c6ae4",
    "tenantId": "62d63eb5a2cf20503b23c2ce",
    "creationDateTime": "2022-08-09T19:51:49.155Z",
    "revisedDateTime": "2022-08-09T19:51:49.155Z",
    "usedFor": "_",
    "metadata": {
      "k": "v"
    },
  },
  "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
}
]
}

```

The following are the description of the response parameters of the *Get list of all immediate child elements for a parent folder API*:

Parameter	Description
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.

Parameter	Description
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform.¶
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination matches with applied tags in the securityClass field. OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Getting a folder metadata by ID

The *Get folder metadata by ID* API allows you to fetch the metadata associated with a folder by its valid ID.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders/metadata/{folderId}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get folder metadata by ID* API, pass the value of the `folderId` parameter in the request URL. For example, if you want to get metadata of a folder whose ID is `630e68a6dfd49e5ebe55167d`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice/folders/metadata/630e68a6dfd49e5ebe55167d
```

The following is the description of the request parameter of the *Get folder metadata by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	string	The ID of the folder whose metadata information is required.	URL

Sample response

```
{
  "<metadata field name>": "<metadata field value>"
}
```

The following is the description of the response parameter of the *Get folder metadata by ID* API:

Parameter	Description
metadata	It is a JSON key-value pair of the metadata to associate with a folder. You can create any custom metadata fields as per your requirements.

Getting a folder data class by ID

The *Get folder data class by ID* API allows you to fetch the data class associated with a folder by its valid ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/folders/dataclass/{folderId}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get folder data class by ID* API, pass the value of the *folderId* parameter in the request URL. For example, if you want to get the data class of a folder whose ID is 630e65b5dfd49e5ebe551674, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/folders/dataclass/630e65b5dfd49e5ebe551674

The following is the description of the request parameter of the *Get folder data class by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	string	The ID of the folder to get its data class.	URL

Sample response

```
{
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "dataClassName": "dataclass_190fg33ss124",
  "age": "20"
}
```

The following are the description of the response parameter of the *Get folder data class by ID* API:

Parameter	Description
dataClass	It is a JSON key-value pair of the data class to associate with a folder. You can create any custom data class fields as per your requirements. For example, the dataClassId, dataClassName, and age are custom-created data class fields in the above sample response.

Getting a folder by ID

The *Get folder by ID* API allows you to get the details of a folder using its folder ID.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders/{{id}}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get folder by ID* API, pass the value of the *id* parameter in the request URL. For example, if you want to get a folder whose ID is 630e65b5dfd49e5ebe551674, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice/folders/
630e65b5dfd49e5ebe551674
```

The following is the description of the request parameter of the *Get folder by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the required folder.	URL

Sample response

```
{
  "id": "630e65b5dfd49e5ebe551674",
  "folderName": "Marketing",
  "folderType": "folder",
  "comments": "Workspace for project A",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "ownerName": "Manoj",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "creationDateTime": "2022-08-30T19:32:05.748Z",
  "revisedDateTime": "2022-08-30T19:32:05.748Z",
  "usedFor": "general",
  "dataclass": {
    "dataClassName": "dataclass_marketing",
    "dataClassId": "62d7f3c3c0ea8e46159b1040"
  },
  "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4",
  "securityClass": [
    "OWNER"
  ]
}
```

The following is the description of the request parameter of the *Get folder by ID* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.

Parameter	Description
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents that match with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Copying a folder

The *Copy folder* API allows you to copy a folder or sub-folder and paste it into another folder.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders/copy/{{folderId}}/{{folderId1}}`
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Copy folder* API, pass the value of the *folderId* and *folderId1* parameters in the request URL. For example, if you want to copy a folder whose ID is 630e6614dfd49e5ebe551675 and paste it to another folder whose ID is 62d63ec46367977f6d813e39, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice//folders/copy/630e6614dfd49e5ebe551675/62d63ec46367977f6d813e39
```

The following are the description of the request parameters of the *Copy folder* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	ID of the folder that needs to be copied.	URL
folderId1	Mandatory			Yes	String	ID of the destination folder.	URL

Sample response

```
{
  "id": "630e6614dfd49e5ebe551675",
  "folderName": "Marketing",
  "folderType": "folder",
  "comments": "Workspace for project A",
  "parentFolderId": "62d63ec46367977f6d813e39",
  "ownerName": "Manoj",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "creationDateTime": "2022-08-30T19:33:40.530Z",
  "revisedDateTime": "2022-08-30T19:36:51.589Z",
  "accessDateTime": "2022-08-30T19:36:51.589Z",
  "usedFor": "general",
  "dataclass": {
    "dataClassName": "dataclass_marketing",
    "dataClassId": "62d7f3c3c0ea8e46159b1040"
  },
  "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4",
  "securityClass": [
    "OWNER"
  ]
}
```

The following are the description of the response parameters of the *Copy folder* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.

Parameter	Description
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform. [¶]
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents that matches with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Checking copy folder status

The *Check copy folder status* API provides the status of the folder whether the folder is uploaded to the cloud (Azure) or not, if uploaded the status appears as completed.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders/copystatus?token={{token}}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Check copy folder status* API, pass the value of the *token* query parameters in the request URL. For example, if you want to check a folder copy status whose token value is `eyJ0eXAiOiJKVlQiL`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice/folders/copystatus?token=eyJ0eXAiOiJKVlQiL
```

The following is the description of the request parameters of the *Check copy folder status* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
token	Mandatory			Yes	String	Value of the token parameter received as a response of copied folder request.	Query params

Sample response

```
{
  "status": "COMPLETED"
}
```

The following is the description of the response parameters of the *Check copy folder status* API:

Parameter	Description
status	It indicates the copied status of the folder.

Moving a folder to trash

The *Move folder to trash* API allows you to move a folder into the trash. The target folder where the required folder has to be moved must be available in the trash.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/moveToTrash/{id}/{targetFolderId}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Move folder to trash* API, pass the value of the *id* and *targetFolderId* parameters in the request URL. For example, if you want to move a folder whose ID is 630e6614dfd49e5ebe551675 into trash folder whose ID is 62d63ec46367977f6d813e39, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/folderservice/folders/moveToTrash/630e6614dfd49e5ebe551675/62d63ec46367977f6d813e39

The following are the description of the response parameters of the *Move folder to trash* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	Current folder ID.	URL
targetFolderid	Mandatory			Yes	String	Trash folder ID.	URL

Sample response

```
{
  "id": "630e6614dfd49e5ebe551675",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Documents related to banking",
  "parentFolderId": "62d63ec46367977f6d813e39",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858307838,
  "accessDateTime": 1661858307838,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
  "dataclass": {
    "dataClassName": "dataclass_banking",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age": "20"
  },
  "primaryContentId": "630ddd693cf0cb412311e33c",
  "latest": true,
}
```

```

"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20",
"accessCount": 0,
"inTrash": true,
"indexed": true,
"parentHierarchy": {
  "folder": [
    {
      "comments": "System Folder - Guide Folder",
      "odId": 0,
      "parentFolderId": "62d63ec46367977f6d813e39",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",
      "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
      "folderName": "Explore Sample Application",
      "id": "62d63ec46367977f6d813e3c",
      "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
    },
    {
      "comments": "System Folder - Tenant Asset Folder",
      "odId": 0,
      "parentFolderId": "62d63ec328268c6a3ee6ba5b",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",
      "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
      "folderName": "Tenant Asset Folder",
      "id": "62d63ec46367977f6d813e39",
      "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
    }
  ],
  "cabinet": {
    "comments": "cabinet created by system",
    "odId": 0,
    "indexed": false,
    "inTrash": false,
    "ownerId": "System",
    "accessDateTime": "Tue Jul 19 05:18:59 GMT 2022",

```

```

    "version": 1,
    "folderType": "cabinet",
    "ownerName": "System",
    "tenantId": "62d63eb5a2cf20503b23c2ce",
    "usedFor": "general",
    "revisedDateTime": "Tue Jul 19 05:18:59 GMT 2022",
    "folderName": "DefaultCabinet",
    "id": "62d63ec328268c6a3ee6ba5b",
    "creationDateTime": "Tue Jul 19 05:18:59 GMT 2022"
  }
},
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Move folder to trash* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.

Parameter	Description
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents that matches with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Moving a folder from trash

The *Move folder from trash* API allows you to restore a folder from the trash.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/restore/{{folderId}}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Move folder from trash* API, pass the value of the *folderId* parameter in the request URL. For example, if you want to restore a folder from trash whose ID is 630e6614dfd49e5ebe551675, then pass the request URL as follows:

```
http http or https://<IP address>:<Port>/ecmapi/folderservice/folders/restore/630e6614dfd49e5ebe551675
```

The following are the description of the request parameter of the *Move folder from trash* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	The ID of the folder to restore it from the trash.	URL

Sample response

```
{
  "id": "630e6614dfd49e5ebe551675",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Document related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858406299,
  "accessDateTime": 1661858406299,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
```

```

"documentType": "png",
"documentSize": "246",
"metadata": {
  "key": "v"
},
"dataclass": {
  "dataClassName": "dataclass_banking",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Move folder from trash* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.

Parameter	Description
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform.¶
usedFor	Describes the purpose of the folder.
securityClass	The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents thatmatch matches with applied tags in the securityClass field. OWNER is a system tag, which defines that the owner of the document or folder has full rights. Format: ["OWNER", "<Global Tag>-<Security Class>"]

Moving a folder

The *Move folder* API allows you to move a folder or sub-folder from one folder to another folder.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/move/{{subFolderId2}}/{{subFolderId1}}*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Move folder* API, pass the value of the *subFolderId1* and *subFolderId2* parameters in the request URL. For example, if you want to move a folder from a sub-folder whose ID is 630e6614dfd49e5ebe551675 to another sub-folder whose ID is 62d63ec46367977f6d813e39, then pass the request URL as follows:

Example: *http or https://<IP address>:<Port>/ecmapi/folderservice//folders/move/630e6614dfd49e5ebe551675/62d63ec46367977f6d813e39*

The following are the description of the request parameters of the *Move folder* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
subFolderId2	Mandatory			Yes	String	ID of the folder that needs to be moved.	URL
subFolderId1	Mandatory			Yes	String	ID of the destination folder.	URL

Sample response

```
{
  "id": "630e6614dfd49e5ebe551675",
  "folderName": "a",
  "folderType": "folder",
  "comments": "My workspace for project A",
  "parentFolderId": "62d63ec46367977f6d813e39",
  "ownerName": "manoj",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "creationDateTime": "2022-08-30T19:33:40.530Z",
  "revisedDateTime": "2022-08-30T19:36:51.589Z",
  "accessDateTime": "2022-08-30T19:36:51.589Z",
  "usedFor": "general",
  "dataclass": {
    "dataClassName": "dataclass_190fg33ss124",
    "dataClassId": "62d7f3c3c0ea8e46159b1040"
  }
}
```

```

},
"lastModifiedByUserId": "62d63eb7016b20476a1c6ae4",
"securityClass": [
  "OWNER"
]
}

```

The following are the description of the response parameters of the *Move folder* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform. ^[1]
usedFor	Describes the purpose of the folder.

Parameter	Description
securityClass	<p>The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents that match with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Searching folder on data class

The *Search folder on data class* API allows you to search a folder based on the data class fields. It is mandatory that the data class field must be associated with the folder.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/dataclass*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "dataClassId": "630dd9103cf0cb412311e33a",
  "dataClassName": "DataClass_Banking"
}
```

The table below describes the request parameters of the *Search folder on data class* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
dataClass	Mandatory			Yes	String	It is a JSON key-value pair of the data class to associate with the folder. You can create any custom data class fields as per your requirements. For example, in the above sample request, the dataClassId and dataClassName are custom-created data class fields.	Request body

Sample response

```
{
  "id": "630e65b5dfd49e5ebe551674",
  "folderName": "Marketing",
  "folderType": "folder",
  "comments": "Workspace for project A",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "ownerName": "Manoj",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "creationDateTime": "2022-08-30T19:32:05.748Z",
  "revisedDateTime": "2022-08-30T19:32:05.748Z",
  "usedFor": "general",
  "dataclass": {
```

```

    "dataClassName": "DataClass_Banking",
    "dataClassId": "630dd9103cf0cb412311e33a"
  },
  "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4",
  "securityClass": [
    "OWNER"
  ]
}

```

The following are the description of the request and response parameters of the *Search folder on data class* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.

Parameter	Description
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email obtained while signing up to the NewgenONE Content Cloud platform.¶
usedFor	Describes the purpose of the folder.
securityClass	The securityClass field is an array to capture the tags applied to a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights defined for this combination on the folders and contents that match with applied tags in the securityClass field. OWNER is a system tag, which defines that the owner of the document or folder has full rights. Format: ["OWNER", "<Global Tag>-<Security Class>"]

Search folders based on metadata

The *Search folders based on metadata* API allows you to search a folder using metadata. You can use this API only if the metadata is associated with the folder which you want to search.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders?metadata.<key>=<value>`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Search folder based on metadata* API, pass the value of the search parameter in the request URL. For example, if you want to search a folder with its metadata value as "banking" which is available in your folder repository, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice/folders?
metadata.folder=banking
```

The following is the description of the request parameter of the *Search folder on metadata* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
searchParams	Mandatory			Yes	String	Parameters based on which the content is to be searched.	Query params

Sample response

```
{
  "id": "630dce060fa8b04c913f1f3e",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Document related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dce050fa8b04c913f1f3d",
  "privilege": "INHERITED",
  "creationDateTime": 1661849094001,
  "revisedDateTime": 1661849094102,
```

```

"accessDateTime": 1661849094102,
"version": 1,
"flag": "COMMITTED",
"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"metadata": {
  "folder": "banking"
},
"dataclass": {
  "dataClassName": "dataclass_banking",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"primaryContentId": "630dce060fa8b04c913f1f3e",
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Search folder on metadata* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last accessed.
comments	Comments for the folder.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was created.
folderName	Name of the folder.
folderType	It is a system-operated field. Set the value of this parameter as 'folder', if you are creating a general folder.
id	It indicates the folder ID. It is an optional field only while creating a folder, otherwise, it is a mandatory field in all other API calls.
lastModifiedByUserId	This is a system-operated field to keep the log of the user IDs who have last modified the folder properties.

Parameter	Description
metadata	It is a JSON Key value pair of metadata associated with a folder. It is a mandatory field while adding the metadata to a folder.
ownerId	ID of the user who creates the folder.
ownerName	Name of the user who creates the folder.
parentFolderId	It is a system-operated ID of the parent folder in which the required folder exists. This is a mandatory field in while calling the create folder API.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the folder was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
usedFor	Describes the purpose of the folder.
securityClass	<p>The securityClass field is an array to capture the tags applied on a folder. A combination of global tag and securityClass are defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and contents on which this combination match with applied tags in the securityClass field.</p> <p>OWNER is a system tag, which defines that the owner of the document or folder has full rights.</p> <p>Format: ["OWNER", "<Global Tag>-<Security Class>"]</p>

Deleting a folder containing children

The *Delete a folder containing children* API allows you to delete a folder that contains multiple children elements.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/folderservice/folders/{{folderId}}?recursive=true`
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete a folder containing children* API, pass the value of the *folderId* parameter in the request URL. For example, if you want to delete a folder (Containing multiple children elements) whose ID is *62d63ec46367977f6d813e3c*, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/folderservice/folders/  
62d63ec46367977f6d813e3c?recursive=true
```

The following is the description of the request parameter of the *Delete a folder containing children* API:

Parameter	Mandatory/ Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	The ID of the folder to delete it that contains multiple children elements.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Deleting a folder

The *Delete folder* API allows you to delete a folder from its repository.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folderservice/folders/{{folderId}}*
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete folder* API, pass the value of the *folderId* parameter in the request URL. For example, if you want to delete a folder whose ID is 62d63ec46367977f6d813e3c, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/folderservice/folders/62d63ec46367977f6d813e3c

The following is the description of the request parameter of the *Delete folder* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	The ID of the folder to delete it.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Deleting a folder data class by ID

The *Delete folder data class by ID* API allows you to delete the associated data class of a folder using a valid folder ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/folders/folderservice/dataclass/{folderId}*
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the Delete folder data class by ID API, pass the value of the folderId parameter in the request URL. For example, if you want to delete the data class of a folder whose ID is 62d63ec46367977f6d813e3c, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/folders/dataclass/62d63ec46367977f6d813e3c

The following is the description of the request parameter of the *Delete folder data class by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	The ID of the folder to delete its data class.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Deleting a folder metadata by ID

The *Delete folder metadata by ID* API allows you to delete the metadata of a folder using its folder ID.

- **Service URL:** http or https://:/ecmapi/folderservice/folders/metadata/{folderId}
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete folder metadata by ID* API, pass the value of the *folderId* parameter in the request URL. For example, if you want to delete the metadata of a folder whose ID is 62d63ec46367977f6d813e3c, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/folderservice/metadata/62d63ec46367977f6d813e3c

The following is the description of the request parameter of the *Delete folder metadata by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
folderId	Mandatory			Yes	String	The ID of the folder to delete its metadata.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Error codes

The following table describes the error codes and their description related to folder services:

Error code	Error message
24001	Parent folder ID cannot be blank.
24002	Invalid folder type.
24003	Source folder not found.
24004	The target folder is a child of the source folder.
24005	No metadata associated with the requested folder.
24006	Target folder not in trash.
24007	Source folder not in trash.
24008	Copy folder not found.
24009	Invalid folder name.
240010	Async copy folder operation not found.
240011	Child folder not found.
240012	Counter not found.
240013	No dataclass associated with the requested folder.
240014	Folder cannot be moved within itself.
240015	Folder cannot be moved to the same location.
240016	Folder type value mismatched.
240017	A trash folder or a subfolder of a trash folder is the target folder.
240018	System-generated folders cannot be moved.
240019	Source folder ID cannot be blank or contain special characters and alphabets only.
240020	Target Folder ID cannot be blank or contain special characters and alphabets only.

Error code	Error message
240021	Folder ID cannot be blank or contain special characters and alphabets only.
240022	Invalid Page Number or Page Number must be greater than zero.
240023	Folder cannot be moved to its subfolder.
240024	DTO is null or value is greater than Max Size.
240025	Parent folder is in the trash.

Content services

Content service deals with documents. A document is a form of information. It may consist of one or more related pages. Documents can be organized in a folder or repository. You can also perform certain operations related to metadata and indexing on the document. Documents can be classified as follows:

Image documents — Image documents are generally scanned documents like BMP, JPG, Tiff, GIF, TIF, and so on.

Non-image documents — Non-image documents are generally Word-processed documents, presentations, spreadsheets, and so on, generated through various applications.

Examples:

- Organizations can use content services for storing all kinds of content generated or dealt with in an enterprise environment. These include text files, documents, slideshows, images, media files such as audio and videos, as well as conversations.
- Store content along with their corresponding metadata such as source application name, agent name, timestamps, and so on.

The content services include the following APIs:

- [Move content to target folder](#)
- [Copy content](#)
- [Search content on metadata](#)
- [Check in content sync](#)

- Get content data class by ID
- Get content metadata by ID
- Get content root path by ID
- Sort contents based on revised date time
- Search content on data class
- Search for largest version
- Upload file
- Get all content
- Get content status
- Retrieve content
- Fetch content model
- Delete content
- Delete content by parent
- Delete content metadata by ID
- Delete content data class by ID
- Move content to trash
- Move content from trash
- Mark content in trash
- Update parent folder with respect to trash content
- Modify a content property
- Content checked out
- Content undo checked out
- Set content version as latest

Headers

The below table lists the header parameters for content services:

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/ json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
userId	Conditional		String	This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed. Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.
appld	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Moving content to target folder

The *Move content to target folder* API allows you to move a document from one folder to another folder.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/move/{{contentId}}/{{subFolderId}}*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Move content to target folder* API, pass the value of the *contentId* parameter in the request URL. For example, if you want to move the content whose ID is 630e6614dfd49e5ebe551675 to a folder whose ID is 630e6614dfd49e5abc551789, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/move/630e6614dfd49e5ebe551675/630e6614dfd49e5abc551789
```

The following are the description of the request and response parameters of *Move content to target folder* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
contentId	Mandatory			Yes	String	The content ID that needs to be moved.	URL
subFolderId	Mandatory			Yes	String	ID of the destination folder.	URL

Sample response

```
{
  "id": "630e6614dfd49e5ebe551675",
  "name": "Invoice",
  "contentType": "Image",
  "parentFolderId": "630e6614dfd49e5abc551789",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dc5e30fa8b04c913f1f38",
  "privilege": "INHERITED",
  "creationDateTime": 1661847011480,
  "revisedDateTime": 1661847388378,
```

```

"accessDateTime": 1661847388378,
"version": 1,
"flag": "COMMITTED",
"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"dataclass": {
  "dataClassName": "dataclass_invoice",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_invoice 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "invoice",
"docSize": 246,
"contentPath": "/var/tmp/e1e57fef-d750-4bb1-b40b-1fe04eac2e91:birthday.png"
}

```

The following are the description of the response parameters of the *Move content to target folder* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.

Parameter	Description
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.

Parameter	Description
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Copying content

The *Copy content* API allows you to copy and paste a document from one folder to another folder.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/copy/{{id}}?{{parentFolderId}}`
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Copy content* API, pass the value of the `id` and `parentFolderId` parameters in the request URL. For example, if you want to copy a document whose ID is `630dc8560fa8b04c913f1f3c` to a folder whose ID is `62d63ec46367977f6d813e3c`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/copy/630dc8560fa8b04c913f1f3c?62d63ec46367977f6d813e3c
```

The following are the description of the request parameters of the *Copy content* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	The ID of the content.	URL params
parentFolderId	Mandatory			Yes	String	The ID of the parent folder.	Query params

Sample response

```
{
  "id": "630dc8560fa8b04c913f1f3c",
  "name": "Birthday",
  "contentType": "Image",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dc5e30fa8b04c913f1f38",
  "creationDateTime": 1661847638016,
  "revisedDateTime": 1661847638048,
  "accessDateTime": 1661847638048,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "dataclass": {
    "dataClassName": "dataclass_birthday",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age": "20"
  },
  "primaryContentId": "630dc8560fa8b04c913f1f3c",
  "latest": true,
  "checkedOut": false,
  "accessCount": 0,
  "inTrash": false,
  "indexed": false,
  "docSize": 0
}
```

The following are the description of the response parameters of the *Copy content* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.

Parameter	Description
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.

Parameter	Description
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.

Parameter	Description
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Searching content on metadata

The *Search content on metadata* API allows you to search the content using metadata. You can use this API only if the metadata is associated with the content which you want to search.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents?metadata.<key>=<value>*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Search content on metadata* API, pass the value of the metadata's key and value in the search parameter in the request URL. For example, if you want to search the banking document available in your document repository with associated metadata, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents?
metadata.document=banking
```

The following is the description of the request parameter of *Search content on metadata* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
searchParams	Mandatory			Yes	String	Parameters based on which the content is to be searched.	Query params

Sample response

```
{
  "id": "630dce060fa8b04c913f1f3e",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Document related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dce050fa8b04c913f1f3d",
  "privilege": "INHERITED",
  "creationDateTime": 1661849094001,
  "revisedDateTime": 1661849094102,
  "accessDateTime": 1661849094102,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
  "dataclass": {
    "dataClassName": "dataclass_banking",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age1": "20"
  },
  "primaryContentId": "630dce060fa8b04c913f1f3e",
```

```

"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of *Search content on metadata* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.

Parameter	Description
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.

Parameter	Description
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Checking-in content sync

The *Check in content sync* API allows you to update a specific version of the document. You must check out the document, once the version is updated.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/content/syn*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "name": "Aug_bill",
  "contentType": "pdf",
  "comments": "Bill of august month",
  "parentFolderId": "630dce060fa8b04c913f1f3e",
  "noOfPages": 1,
  "documentSize": 12,
  "documentType": "pdf",
  "storageCredentialId": "62b9533e56e54b30b1e5494f",
  "primaryContentId": "630dce060fa8b04c913f1f3e",
  "versioningType": "minor",
  "file": "Bill"
}
```

The following are the description of the request parameters of the *Check in content sync* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
file	Mandatory				Multipart File	Document or file that needs to be uploaded. You can upload multiple files. It is a mandatory field to create content.	Request body
name	Mandatory				String	Name of the document.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
content Type	Mandatory			Yes	String	It is a system operated field. If you are uploading general content in pdf form, then the content type is 'pdf'. Similarly, if you are uploading content of type 'jpg (picture)', then the content type is 'png'.	Request body
primary ContentId	Mandatory			Yes	String	It is an ID of content.	Request body
versioning Type	Mandatory			Yes	String	Indicates minor or major depending upon versioning.	Request body
checkedIn byUser Name	Optional				String	Checked-in user name.	Request body
dataclass	Optional				String	A JSON Key value pair of the data indexing associated with the Folder.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
metadata	Conditional				String	A JSON Key value pair of metadata associated with the Folder. Mandatory while using API to add metadata to a folder.	Request body
privilege	Conditional	INHERITED, PRIVATE, SHARED	INHERITED		String	The privilege of the user who is accessing the folder ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the folder is created as private or shared.	Request body
noOf Pages	Mandatory				String	The number of pages in the uploaded file.	Request body
document Type	Mandatory				String	Extension of the documents. For example, pdf, doc, xls, png, jpeg, and so on.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
document Size	Mandatory				String	The size of a document	Request body
storage Credential Id	Mandatory			Yes	String	This is a system-level field. This field is passed in the response of the user login API, you can copy and paste the credential ID from the user login API.	
comments	Optional				String	Comments for the folder.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
security Class	Optional	Example: ["OWNER", "Department-topsecret"]		Yes	String array	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.	Request body

Sample response

```
{
  "id": "630dd9103cf0cb412311e33a",
  "name": "Aug_bill",
  "contentType": "pdf",
  "comments": "August month bill",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dd9103cf0cb412311e339",
  "privilege": "INHERITED",
  "creationDateTime": 1661849094001,
  "revisedDateTime": 1661851920213,
  "accessDateTime": 1661851920213,
  "version": 1.1,
  "previousVersion": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "pdf",
  "documentSize": "246",
  "primaryContentId": "630dce060fa8b04c913f1f3e",
  "latest": true,
  "checkedOut": false,
  "lastCheckedInBy": "62d63eb7016b20476a1c6ae4",
  "previousVersionContentId": "630dce060fa8b04c913f1f3e",
  "accessCount": 0,
  "inTrash": false,
  "indexed": false,
  "nameLowerCase": "aug_bill",
  "docSize": 246
}
```

The following are the description of the response parameters of the *Check in content sync* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.

Parameter	Description
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.

Parameter	Description
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.

Parameter	Description
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Getting content metadata by ID

The *Get content metadata by ID* API allows you to search the metadata of a document using its content ID if the metadata is associated with the document.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/metadata/{contentId}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get content metadata by ID* API, pass the value of the *contentId* parameters in the request URL. For example, if you want to search the metadata associated with a document whose document ID is 630dd9103cf0cb412311e33a, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/metadata/630dd9103cf0cb412311e33a
```

The following is the description of the request parameter of the *Get content metadata by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
contentId	Mandatory			Yes	String	The ID of the content to be searched.	URL

Sample response

```
{
  "metadataFieldName": "metadataFieldValue"
}
```

The following is the description of the response parameter of the *Get content metadata by ID* API:

Parameter	Description
metadata	It is a JSON key-value pair of the metadata to associate with the content. You can create any custom metadata fields as per your requirements.

Getting content data class by ID

The *Get content data class by ID* API allows you to search a data class using its content ID if the data class is associated with the document.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/content/dataclass/{contentId}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get content data class by ID* API, pass the value of the *contentId* parameters in the request URL. For example, if you want to search a data class associated with a document whose document ID is 630dd9103cf0cb412311e33a, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/dataclass/630dd9103cf0cb412311e33a

The following is the description of the request parameter of the *Get content data class by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
contentId	Mandatory			Yes	String	The ID of the content.	URL

Sample response

```
{
  "dataClassId": "630dd9103cf0cb412311e33a",
  "dataClassName": "DataClass_Banking",
  "age": "20"
}
```

The following is the description of the response parameter of the *Get content data class by ID* API:

Parameter	Description
dataClass	It is a JSON key-value pair of the data class to associate with the content. You can create any custom data class fields as per your requirements. For example, in the above sample response the <i>dataClassId</i> , <i>dataClassName</i> , and <i>age</i> are custom-created data class fields.

Getting content root path by ID

The *Get content root path by ID* API allows you to fetch the root path hierarchy of a document using its content ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/rootpathhierarchy/{contentId}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get content root path by ID* API, pass the value of the *contentId* parameter in the request URL. For example, if you want to get the root path for content whose ID is *Get content root path by ID*, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/rootpathhierarchy/62d63ec46367977f6d813e3c

The following is the description of the request parameter of the *Get content root path by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
contentId	Mandatory			Yes	String	The ID of the content to be searched.	URL

Sample response

```
{
  "folder": [
    {
      "comments": "System Folder - Guide Folder",
      "odId": 0,
      "parentFolderId": "62d63ec46367977f6d813e3c",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",
      "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
      "folderName": "Explore Sample Application",
      "id": "62d63ec46367977f6d813e3c",
      "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
    },
    {
      "comments": "System Folder - Tenant Asset Folder",
      "odId": 0,
      "parentFolderId": "62d63ec328268c6a3ee6ba5b",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",
      "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
      "folderName": "Tenant Asset Folder",
      "id": "62d63ec46367977f6d813e39",
      "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
    }
  ],
  "cabinet": {
    "comments": "cabinet created by system",
    "odId": 0,
    "indexed": false,
    "inTrash": false,
    "ownerId": "System",
  }
}
```

```

"accessDateTime": "Tue Jul 19 05:18:59 GMT 2022",
"version": 1,
"folderType": "cabinet",
"ownerName": "System",
"tenantId": "62d63eb5a2cf20503b23c2ce",
"usedFor": "general",
"revisedDateTime": "Tue Jul 19 05:18:59 GMT 2022",
"folderName": "Default Cabinet",
"id": "62d63ec328268c6a3ee6ba5b",
"creationDateTime": "Tue Jul 19 05:18:59 GMT 2022"
},
"content": {
  "checkedOut": false,
  "primaryContentId": "630dce060fa8b04c913f1f3e",
  "metadata": {
    "key": "v"
  },
  "flag": "COMMITTED",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "dataclass": {
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "dataClassName": "dataclass_system",
    "age": "20"
  },
  "documentType": "png",
  "privilege": "INHERITED",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "documentSize": "246",
  "accessDateTime": "Tue Aug 30 09:32:00 GMT 2022",
  "ownerName": "James",
  "noOfPages": "1",
  "id": "630dce060fa8b04c913f1f3e",
  "contentType": "pdf",
  "contentLocationId": "630dce050fa8b04c913f1f3d",
  "latest": false,
  "creationDateTime": "Tue Aug 30 08:44:54 GMT 2022",
  "securityClass": [
    "OWNER"
  ],
  "accessCount": 0,
  "comments": "j",
  "indexed": false,
  "nameLowerCase": "system",
  "dataClassText": "dataclass_system 62d7f3c3c0ea8e46159b1040 20 ",
  "docSize": 246,
  "checkedOutBy": "",
  "inTrash": false,
  "version": 1,
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "name": "Banking",
  "revisedDateTime": "Tue Aug 30 09:32:00 GMT 2022"
}

```



```
}
}
```

The following are the description of the response parameters of the *Get content root path by ID* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.

Parameter	Description
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.

Parameter	Description
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Sorting content based on revised date time

The *Sorting content based on revised date time* API allows you to sort the latest content based on the revised data and time.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/getLastModifiedContent/{parentFolderId}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Sorting content based on revised date time* API, pass the value of the *parentFolderId* parameter in the request URL. For example, if you want to sort the latest version of a document whose ID is 62d63ec46367977f6d813e39, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/
getLastModifiedContent/62d63ec46367977f6d813e39
```

The following is the description of the request parameter of the *Sorting content based on revised date time* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
parentFolderId	Mandatory			Yes	String	The ID of the parent folder to sort the content.	URL

Sample response

```
{
  "id": "62d63ec46367977f6d813e39",
  "name": "Birthday",
  "contentType": "Image",
  "parentFolderId": "62d63ec46367977f6d813e39",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dc5e30fa8b04c913f1f38",
  "privilege": "INHERITED",
  "creationDateTime": 1661847011480,
  "revisedDateTime": 1661847388378,
  "accessDateTime": 1661847388378,
  "version": 1,
  "flag": "COMMITTED",
```

```

"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"dataclass": {
  "dataClassName": "dataclass_birthday",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age1": "20"
},
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_birthday 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "birthday",
"docSize": 246,
"contentPath": "/var/tmp/e1e57fef-d750-4bb1-b40b-1fe04eac2e91:birthday.png"
}

```

The following are the description of the response parameters of the *Sorting content based on revised date time* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.

Parameter	Description
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.

Parameter	Description
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Searching content based on data class

The *Search content on data class* API allows you to search content based on the data class fields. It is mandatory that the data class field must be associated with the content.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/dataclass*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "dataClassId": "630dd9103cf0cb412311e33a",
  "dataClassName": "DataClass_Banking",
  "age": "20"
}
```


The following is the description of the request parameter of the *Search content on data class* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
dataClass	Mandatory			Yes	String	It is a JSON key-value pair of the data class to associate with the content. You can create any custom data class fields as per your requirements. For example, in the above sample request the dataClassId, dataClassName, and age are custom-created data class fields.	Request body

Sample response

```
{
  "id": "62fb708026343069bf046da1",
  "usedFor": null,
  "name": "Finance",
  "contentType": "pdf",
  "comments": "Document related to finance",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "62fb708026343069bf046da0",
  "creationDateTime": 1660645504523,
  "revisedDateTime": 1660645504598,
}
```

```

"accessDateTime": 1660645504598,
"version": 1,
"previousVersion": null,
"flag": "COMMITTED",
"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"metadata": null,
"dataclass": {
  "dataClassName": "DataClass_Banking",
  "dataClassId": "630dd9103cf0cb412311e33a",
  "age": "20"
},
"primaryContentId": "62fb708026343069bf046da1",
"latest": true,
"versionComments": null,
"checkedOut": false,
"checkedOutBy": null,
"lastCheckedInBy": null,
"lastCheckedInbyUserName": null,
"previousVersionContentId": null,
"checkedOutTime": null,
"parentHierarchy": null,
"docSize": 246
}

```

The following is the description of the response parameter of the *Search content on data class* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.

Parameter	Description
inTrash	It indicates whether the content or folder is in the trash or not.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.

Parameter	Description
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Searching for largest version

The *Searching for largest version* API allows you to search for the latest version of a document.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/fetchLargestVersion/{primaryContentId}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Searching for largest version* API, pass the value of the `primaryContentId` parameter in the request URL. For example, if you want to search the latest version of a document whose ID is `62d63ec46367977f6d813e39`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/fetchLargestVersion/62d63ec46367977f6d813e39
```

The following is the description of the request parameter of the *Searching for largest version* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
primaryContentId	Mandatory			Yes	String	The ID of the content to be searched.	URL

Sample response

```
{  
  "largestVersion": 1.1  
}
```

The following is the description of the response parameter of the *Searching for largest version* API:

Parameter	Description
largestVersion	This parameter indicates the latest version of a document.

Uploading a file

Uploading a file contains:

- [Generating Blob URI](#)
- [Uploading to Blob.](#)

Generating Blob URI

The *Generating Blob URI* API allows you to upload documents of any size. It also allows you to read, play, and upload multimedia files, including text, audio, images, and video.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/content/generateFileUri*
- **Method Type:** POST
- **Request Content Type:** application/json
- **Response Content Type:** application/json

Sample request

While calling the *Generating Blob URI* API, pass the value of the *id* parameter in the request URL. For example, if you want to upload a file, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/generateFileUri

```
{
  "name": "ECM Next Platform - Services Details",
  "storageCredentialId": "64702b530ed8915e26330127",
  "parentFolderId": "6476d38f0dfdec776932f356",
  "ownerName": "Himanshu Kumar",
  "ownerId": "64702ad33422f7200041e9de",
  "usedFor": "sjeqwe s",
  "securityClass": [
    "OWNER"
  ],
  "documentSize": 216099,
  "documentType": "xlsx",
  "contentType": "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet"
}
```

The following table describes the request parameter of the *Generating Blob URI* API:

Parameters	Mandatory/Optional	Accepted Values	Default Values	System Field	Data Type	Description
parentFolderId	Yes			Yes	string	Indicates the system-generated ID of a parent or root folder.
ownerId	Yes				string	Represents the user ID who creates the folder.
uploadIn	No			Yes	string	Describes the location where the file gets added. For example, Repository.
usedFor	Optional				String	Describes the purpose of the folder.

Parameters	Mandatory/Optional	Accepted Values	Default Values	System Field	Data Type	Description
dataclass	Optional				String	<p>Represents the JSON key-value pair of the data indexing associated with the folder. format:</p> <pre> { "dataClassName": " ", "dataClassId": " ", "dataClassField Name": " ", "dataClassInteger FieldName": , "dataClassArray Field Name": [" ", " "] } </pre>
metadata	Conditional <ul style="list-style-type: none"> It is Mandatory while using API to add metadata in a folder. 				String	Represents the JSON key-value pair of the metadata associated with the folder.
ownerName	Yes					Represents the username who creates the folder.

Parameters	Mandatory/Optional	Accepted Values	Default Values	System Field	Data Type	Description
noOfPages	Conditional				String	Represents the number of pages in the uploaded file. <ul style="list-style-type: none"> There is only PDF support format for the file.
document Type	Yes				String	Indicates the extension of the document, such as pdf, doc, xls, png, jpg, and more.
document Size	Yes			Yes	String	Indicates the size of a document.
storage CredentialId	Yes			Yes	String	Specifies the field that you must provide in the when the user signs-in.
comments	Optional				String	Represents the comments for the folder.
security Class	Optional	For example, [OWNER, Department-topsecret]		Yes	String array	Represents an array that captures the tags applied to a document.

Sample response

```
{
  "id": "64917dc4ee5bcd4c6669b7e7",
  "usedFor": "sjeqwe s",
  "sasBlobUri": "https://ecmblob.blob.core.windows.net/cf31825c-c639-4b20-b1ac-805f3df1b624/acf6cac8-467b-49e5-a419-f50378501eefECM%20Next%20Platform%20-%20Services%
```

```

20Details.xlsx?sig=mmRby9ClfQOPxlSuB4s9bBRzEiGhnsphahA
xPgWddWk%3D&st=2023-06-20T10%3A21%3A56Z
&se=2023-06-20T10%3A22%3A56Z&sv=
2019-02-02&spr=https&sp=racw&sr=b",
  "name": "ECM Next Platform - Services Details",
  "contentType": "application/vnd.openxmlformats-
officedocument.spreadsheetml.sheet",
  "parentFolderId": "6476d38f0dfdec776932f356",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "Himanshu Kumar",
  "ownerId": "64702ad33422f7200041e9de",
  "tenantId": "64702ad1c990127c015d850e",
  "creationDateTime": "2023-06-20T10:21:56.431Z",
  "revisedDateTime": 1687256516431,
  "accessDateTime": 1687256516431,
  "version": 1,
  "flag": "COMMITTED",
  "documentType": "xlsx",
  "documentSize": "216099",
  "primaryContentId": "64917dc4ee5bcd4c6669b7e7",
  "checkedOut": false,
  "docSize": 216099
}

```

The following table describes the response parameters for the *Generating Blob URI* API:

Parameters	Description
id	Represents the system-generated ID of the uploaded document.
usedFor	Describes the purpose of the folder.
sasBlobUri	Indicates the URI through which you can upload documents at Microsoft Azure cloud. It also contains the policies such as expiry time, creation date time, and protocols used as per the requirements. For more information, see Uploading to Blob .
name	Represents the name of the document.
contentType	Represents the document content type.
parentFolderId	Represents the system-generated ID for the parent folder.
securityClass	Captures the combination that matches with the applied tags on the document.
ownerName	Represents the user name who created the folder.

Parameters	Description
ownerId	Represents the user ID who created the folder.
tenantId	Represents the tenant ID of the document.
creationDateTime	Represents the document creation date and time.
revisedDateTime	Represents the document revised date and time.
accessDateTime	Represents the document accessed date and time.
version	Represents the latest version of the uploaded document.
flag	Indicates the flag that determines whether the document upload operation is successful that is, COMMITTED.
documentType	Indicates the uploaded document type that is, xls, doc, pdf, and more.
documentSize	Represents the uploaded document size.
primaryContentId	Represents the system-generated ID of the uploaded document. This is similar to the id parameter because both the parameters stores the same value. However, while fetching document IDs in different APIs, it is recommended to use the primaryContentId parameter. It is because the variable ID stores the ID of every document in a collection, which is system generated for every collection in nosql database.
checkedOut	Indicates whether the document has checked out or not.

Uploading to Blob

The *Upload to Blob* API allows you to upload documents of any size.

- **URL:** `<sasBlobUri>`
- **Method type:** PUT
- **Request content type:** `application/octet-stream`
- **Response content type:** `application/json`

Sample request

While calling the *Upload to Blob* API, pass the `sasBlobUri` in the request URL. For example, if you want to upload the content of the file, then pass the request URL as follows:

```
https://ecmblob.blob.core.windows.net/1029c5d9-0486-4134-96a7-ae9cdada62e3/af424556-08dd-4a5cb-72b21ad41d17NewgenONE%20Content%20Cloud%201.2%20Developer%20Guide.pdf?sig=3dlQ40P%2BPuZUa8CKDP6IO52lwkoKGQUMHocVnLbIRzI%3D&st=2023-12-01T05%3A42%3A59Z&se=2023-1
```

 The request body is in *Binary* format and attach the file to upload.

The following table describes the request parameter of the *Upload to Blob* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Description	Parameter Type
Content-Length	Mandatory				Indicates the exact byte length of the HTTP body.	Header
Host	Mandatory				Indicates a Blob storage endpoint that creates a base address for all objects within a storage account.	

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Description	Parameter Type
Origin	Mandatory				Indicates a unicode serialization of the origin in the following URL format: <i>http</i> or <i>https://<Domain>:<Port></i>	

Sample response

HTTP Status Code: 201

HTTP Status Code Value: Created

Getting all content

The *Getting all content* API allows you to get the list of all contents for a particular tenant.

- **Service URL:** *http* or *https://<IP address>:<Port>/ecmapi/contentservice/contents?<SearchQueryParam>*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

Case 1: If the tenant has few documents, then pass the below request URL format:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents?
{HttpServletRequest}
```

Where, *HttpServletRequest* specifies the query parameters, for example, "parentFolderId":"62d63ec46367977f6d813e3c"

Case 2: If the tenant has a large number of documents having 62d63ec46367977f6d813e3c parent folder ID, then pass the request URL as follows:

*http or https://<IP address>:<Port>/ecmapi/contentservice/contents/
62d63ec46367977f6d813e3c*

In case 2 set the header parameters as provided in the below table:

Parameter	Mandatory/ Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
offset	Mandatory			Yes	Integer	This parameter is used to set from where the documents list starts to display.	Request header
limit	Optional				Integer	This parameter is used to set the number of documents displayed in the list. In the case of a large number of documents, it is recommended to pass the value of the limit parameter.	Request header
sortOn	Mandatory				String	This parameter is used to set the field to sort documents displayed in the list. You can sort the documents using revisedDateTime and name fields.	Request header
sortOrder	Mandatory	ASC or DESC			String	This parameter is used to set the documents sorted list in ascending and descending order. Where ASC specifies the ascending order and DESC specifies the descending order.	Request header

Sample response

```
[
  {
    "id": "630ddd693cf0cb412311e33c",
    "usedFor": null,
    "name": "Banking",
    "contentType": "pdf",
    "comments": "Document related to banking",
    "parentFolderId": "62d63ec46367977f6d813e3c",
    "securityClass": [
      "OWNER"
    ],
    "ownerName": "John",
    "ownerId": "62d63eb7016b20476a1c6ae4",
    "tenantId": "62d63eb5a2cf20503b23c2ce",
    "contentLocationId": "630ddd693cf0cb412311e33b",
    "creationDateTime": 1661853033916,
    "revisedDateTime": 1661853033945,
    "accessDateTime": 1661853033945,
    "version": 1,
    "previousVersion": null,
    "flag": "COMMITTED",
    "noOfPages": "1",
    "documentType": "png",
    "documentSize": "246",
    "metadata": {
      "key": "v"
    },
    "dataclass": {
      "dataClassName": "dataclass_banking",
      "dataClassId": "62d7f3c3c0ea8e46159b1040",
      "age": 20
    },
    "primaryContentId": "630ddd693cf0cb412311e33c",
    "latest": true,
    "versionComments": null,
    "checkedOut": false,
    "checkedOutBy": null,
    "lastCheckedInBy": null,
    "lastCheckedInbyUserName": null,
    "previousVersionContentId": null,
    "checkedOutTime": null,
    "parentHierarchy": null,
    "docSize": 246
  },
  {
    "id": "630ddd693cf0cb412311e33c",
    "usedFor": null,
```

```

"name": "Invoice",
"contentType": "pdf",
"comments": "Invoice document",
"parentFolderId": "62d63ec46367977f6d813e3c",
"securityClass": [
  "OWNER"
],
"ownerName": "Roy",
"ownerId": "62d63eb7016b20476a1c6ae4",
"tenantId": "62d63eb5a2cf20503b23c2ce",
"contentLocationId": "630ddd693cf0cb412311e33b",
"creationDateTime": 1661853033916,
"revisedDateTime": 1661853033945,
"accessDateTime": 1661853033945,
"version": 1,
"previousVersion": null,
"flag": "COMMITTED",
"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"metadata": {
  "key": "v"
},
"dataclass": {
  "dataClassName": "dataclass_invoice",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": 20
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"versionComments": null,
"checkedOut": false,
"checkedOutBy": null,
"lastCheckedInBy": null,
"lastCheckedInbyUserName": null,
"previousVersionContentId": null,
"checkedOutTime": null,
"parentHierarchy": null,
"docSize": 246
}
]

```

The following are the description of the response parameters of the *Getting all content* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.

Parameter	Description
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.

Parameter	Description
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.

Parameter	Description
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Getting content status

The *Get content status* API provides you the status of whether the document is uploaded to the cloud (Azure) or not, if uploaded the status appears as completed.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/status*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get content status* API, pass the value of the *id* query parameter in the request URL. For example, if you want to check the document status whose ID is *630dd9103cf0cb412311e33a*, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/status?id=630dd9103cf0cb412311e33a
```

The following is the description of the request parameter of the *Get content status* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	Valid content ID.	Query parameter

Sample response

```
{
  "status": "COMPLETED"
}
```

The following is the description of the response parameter of the *Get content status* API:

Parameter	Description
Status	It indicates the upload status of the content on the Cloud platform such as Azure.

Retrieving content

The *Retrieve content* API allows you to retrieve the content using the below Service URL format and provides the two URLs in response, one is a download URL (Azure) and another one is a streaming URL .

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/retrieve/{id}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Retrieve content* API, pass the value of the *id* parameter in the request URL. For example, if you want to retrieve the content whose ID is 630e6614dfd49e5ebe551675, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/retrieve/630e6614dfd49e5ebe551675
```

The following is the description of the request parameter of the *Retrieve content* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	Valid content ID.	URL
docSize	Mandatory			Yes	String	The size of the document.	Query Param
docType	Mandatory			Yes	String	The type of the document.	Query Param

Sample response

```
{
  "downloadUrl": "https://ecmblob.blob.core.windows.net/6dfcf776-c432-4fff-b1f0-fdf6710e839b/c29fa943-a0b6-4cd3-921f-b452dd33f06f:birthday.png?sig=HjVql5iRzTvZ5A59pACUSfUGZUtAidY1tzGaMevo0aI%3D&sv=2019-02-02&si=heath&sr=c",
  "streamingUrl": "https://ecmblob-cdn.azureedge.net/6dfcf776-c432-4fff-b1f0-fdf6710e839b/c29fa943-a0b6-4cd3-921f-b452dd33f06f:birthday.png?sv=2020-08-04&ss=b&srt=o&sp=rli&se=2022-11-30T18:57:30Z&st=2021-12-09T10:57:30Z&spr=https&sig=8agf%2FEsAqnSs2PZuJKHWkEYypxZYOilD7wt%2F0HB0tEI%3D"
}
```

The following are the description of the response parameters of the *Retrieve content* API:

Parameter	Description
downloadUrl	This parameter provides the URL from where you can download the required document.
streamingUrl	This parameter provides the URL from where you can stream or buffer the required document.

Fetching content properties

The *Fetch content model* API allows you to get the details of content using its content ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/{id}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Fetch content model* API, pass the value of the *id* parameter in the request URL. For example, if you want to fetch details of a document whose ID is 62d63ec46367977f6d813e3c, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/62d63ec46367977f6d813e3c
```

The following is the description of the request parameter of the *Fetch content model* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	Valid content ID.	URL

Sample response

```
{
  "id": "62d63ec46367977f6d813e3c",
  "name": "Aug_bill",
  "contentType": "pdf",
  "comments": "test comments",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "ownerName": "df",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dd9103cf0cb412311e339",
  "privilege": "INHERITED",
  "creationDateTime": 1661849094001,
  "revisedDateTime": 1661851920213,
  "accessDateTime": 1661851920213,
  "version": 1.1,
  "previousVersion": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "pdf",
  "documentSize": "246",
  "primaryContentId": "630dce060fa8b04c913f1f3e",
  "latest": true,
  "checkedOut": false,
  "lastCheckedInBy": "62d63eb7016b20476a1c6ae4",
  "previousVersionContentId": "630dce060fa8b04c913f1f3e",
  "accessCount": 0,
  "inTrash": false,
  "indexed": false,
  "nameLowerCase": "aug_bill",
  "docSize": 246
}
```

The following are the description of the response parameters of the *Fetch content model* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.

Parameter	Description
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.

Parameter	Description
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Deleting content

The *Delete content* API allows you to delete a document using its document ID.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/{{id}}*
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete content* API, pass the value of the *id* parameter in the request URL. For example, if you want to delete a document whose ID is *62d63ec46367977f6d813e3c*, then pass the request URL as follows:

*http or https://<IP address>:<Port>/ecmapi/contentservice/contents/
62d63ec46367977f6d813e3c*

The following is the description of the request parameter of the *Delete content* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the content.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Deleting content by parent

The *Delete content by parent* API allows you to delete the content by its parent folder.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/parent/{id}*
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete content by parent* API, pass the value of the `id` parameter in the request URL. For example, if you want to delete a document whose parent folder ID is `62d63ec46367977f6d813e3c`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/parent/62d63ec46367977f6d813e3c
```

The following is the description of the request parameter of the *Delete content by parent* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the parent folder.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Deleting content metadata by ID

The *Delete content metadata by ID* API allows you to delete the associated metadata with the document by its ID.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/metadata/{contentId}`
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete content metadata by ID* API, pass the value of the *contentId* parameter in the request URL. For example, if you want to delete the metadata associated with a document whose ID is `62d63ec46367977f6d813e3c`, then pass the request URL as follows:

`http or https://<IP address>:<Port>/ecmapi/contentservice/contents/metadata/62d63ec46367977f6d813e3c`

The following is the description of the request parameter of the *Delete content metadata by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
contentId	Mandatory			Yes	String	ID of the content.	URL

Sample response

HTTP Status Code: 204

HTTP Status Code Value: No Content

Deleting content data class by ID

The *Delete content data class by ID* API allows you to delete the associated data class with the document by its ID.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/dataclass/{contentId}`
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete content data class by ID* API, pass the value of the *contentId* parameter in the request URL. For example, if you want to delete the data class associated with a document whose ID is *62d63ec46367977f6d813e3c*, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/dataclass/62d63ec46367977f6d813e3c

The following are the description of the request parameters of the *Delete content data class by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
contentId	Mandatory			Yes	String	ID of the content.	URL

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Moving content from trash

The *Move content from trash* API allows you to restore the content from the trash.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/restore/{{id}}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Move content from trash* API, pass the value of the *id* parameter in the request URL. For example, if you want to restore the content from trash whose ID is 630e6614dfd49e5ebe551675, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/restore/  
630e6614dfd49e5ebe551675
```

The following is the description of the request parameter of the *Move content from trash* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the content to be restored.	URL

Sample response

```
{
  "id": "630e6614dfd49e5ebe551675",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Documents related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858406299,
  "accessDateTime": 1661858406299,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
  "dataclass": {
    "dataClassName": "dataclass_banking",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age": 20
  },
  "primaryContentId": "630ddd693cf0cb412311e33c",
  "latest": true,
  "checkedOut": false,
  "dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
}
```



```

"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Move content from trash* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.

Parameter	Description
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.

Parameter	Description
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Marking content to trash

The *Mark content in trash* API allows you to delete the available content. The deleted content moves into the trash from where you can restore or delete the content permanently.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/markInOrNotInTrash/{id}/{toTrash}`
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Mark content in trash* API, set the value of the *version* in the query parameter and pass the value of the *id* and *toTrash* parameters in the request URL. For example, if you want to mark content in trash whose ID is 62d63ec46367977f6d813e3c, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/markInOrNotInTrash/62d63ec46367977f6d813e3c/true
```

The following are the description of the request parameters of the *Mark content in trash* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of content.	URL params
version	Optional				Big Decimal	Version of the content.	Query params

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
toTrash	Mandatory				String	It is used to move a folder into the trash. Pass "true" if the content is required to be moved to the trash. For any other boolean or non-boolean values, it is considered as a "false" value. No additional message is shown for non "true" values.	URL params

Sample response

```
{
  "id": "630ddd693cf0cb412311e33c",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Upload documents related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858582018,
```

```

"accessDateTime": 1661858582018,
"version": 1,
"flag": "COMMITTED",
"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"metadata": {
  "key": "v"
},
"dataclass": {
  "dataClassName": "dataclass_banking",
  "dataClassId": "62d7f3c3c0ea8e46159b1040",
  "age": "20"
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": true,
"indexed": true,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Mark content in trash* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.

Parameter	Description
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.

Parameter	Description
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.

Parameter	Description
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Updating parent folder with respect to trash content

The *Update parent folder with respect to trash content* API allows you to move the specific content available in trash to another folder which is also available in the trash repository.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/updateParentForTrash/{id}/{targetFolderId}`
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Update parent folder with respect to trash content* API, set the value of the *version* in the query parameter and pass the value of the *id* and *targetId* parameters in the request URL. For example, if you want to move a content within trash whose ID is `56d63ec46367977f6d813w3` into another folder available in the trash whose ID is `62d63ec46367977f6d813e3c`, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/
updateParentForTrash/56d63ec46367977f6d813w3 /62d63ec46367977f6d813e3c
```

The following are the description of the request parameters of the *Update parent folder with respect to trash content* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of content.	URL params
version	Optional				Big Decimal	Version of a content.	Query params
targetId	Mandatory				String	Target folder ID.	URL params

Sample response

```
{
  "id": "56d63ec46367977f6d813w3",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Document related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858582018,
  "accessDateTime": 1661858582018,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
  "dataclass": {
    "dataClassName": "dataclass_invoice",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age": "20"
  }
}
```

```

},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_invoice 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": true,
"indexed": true,
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Update parent folder with respect to trash content* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.

Parameter	Description
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.

Parameter	Description
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Moving content to trash

The *Move content to trash* API allows you to move a folder into the trash. The target folder where the required folder has to be moved must be available in the trash.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/moveToTrash/{id}/{targetFolderId}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Move content to trash* API, pass the value of the *id* and *targetFolderId* parameters in the request URL. For example, if you want to move a content whose ID is 630e6614dfd49e5ebe551675 into trash folder whose ID is 62d63ec46367977f6d813e39, then pass the request URL as follows:

`http or https://<IP address>:<Port>/ecmapi/contentservice/contents/moveToTrash/630e6614dfd49e5ebe551675/62d63ec46367977f6d813e39`

The following are the description of the request parameters of the *Move content to trash* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	Current content ID.	URL
targetFolderId	Mandatory			Yes	String	Trash folder ID.	URL

Sample response

```
{
  "id": "630ddd693cf0cb412311e33c",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Document related to banking",
  "parentFolderId": "62d63ec46367977f6d813e3a",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858307838,
  "accessDateTime": 1661858307838,
  "version": 1,
  "flag": "COMMITTED",
}
```

```

"noOfPages": "1",
"documentType": "png",
"documentSize": "246",
"metadata": {
  "key": "v"
},
"dataclass": {
  "code": "500",
  "message": "Exception sending message; nested exception is
com.mongodb.MongoSocketWriteException: Exception sending message"
},
"primaryContentId": "630ddd693cf0cb412311e33c",
"latest": true,
"checkedOut": false,
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20 ",
"accessCount": 0,
"inTrash": true,
"indexed": true,
"parentHierarchy": {
  "folder": [
    {
      "comments": "System Folder - Guide Folder",
      "odId": 0,
      "parentFolderId": "62d63ec46367977f6d813e39",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",
      "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
      "folderName": "Explore Sample Application",
      "id": "62d63ec46367977f6d813e3c",
      "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
    },
    {
      "comments": "System Folder - Tenant Asset Folder",
      "odId": 0,
      "parentFolderId": "62d63ec328268c6a3ee6ba5b",
      "indexed": false,
      "inTrash": false,
      "privilege": "INHERITED",
      "ownerId": "System",
      "version": 1,
      "folderType": "folder",
      "ownerName": "System",
      "tenantId": "62d63eb5a2cf20503b23c2ce",
      "usedFor": "general",

```

```

    "revisedDateTime": "Tue Jul 19 05:19:00 GMT 2022",
    "folderName": "TenantAssetFolder",
    "id": "62d63ec46367977f6d813e39",
    "creationDateTime": "Tue Jul 19 05:19:00 GMT 2022"
  }
],
"cabinet": {
  "comments": "cabinet created by system",
  "odId": 0,
  "indexed": false,
  "inTrash": false,
  "ownerId": "System",
  "accessDateTime": "Tue Jul 19 05:18:59 GMT 2022",
  "version": 1,
  "folderType": "cabinet",
  "ownerName": "System",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "usedFor": "general",
  "revisedDateTime": "Tue Jul 19 05:18:59 GMT 2022",
  "folderName": "DefaultCabinet",
  "id": "62d63ec328268c6a3ee6ba5b",
  "creationDateTime": "Tue Jul 19 05:18:59 GMT 2022"
}
},
"nameLowerCase": "banking",
"docSize": 246
}

```

The following are the description of the response parameters of the *Move content to trash* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.

Parameter	Description
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.

Parameter	Description
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.

Parameter	Description
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Modifying content properties

The *Modify content properties* API allows you to update the content or document fields.

- **Service URL:** `http or https://<IP address>:<Port>/ecmapi/contentservice/contents/{{id}}`
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Modify content properties* API, pass the value of the *id* parameter in the request URL. For example, if you want to modify the properties of a document whose ID is 630ddd693cf0cb412311e33c, then pass the request URL as follows:

```
http or https://<IP address>:<Port>/ecmapi/contentservice/contents/630ddd693cf0cb412311e33c
```

Request body:

```
{
  "<updateContentParams>": "<Value of the folder parameter that needs to be updated>"
}
```

The following are the description of the request parameters of the *Modify a content properties* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of a content.	URL
<update Content Params>	Mandatory			Yes	String	<p>It is used to have the updated values of content. All fields sent as part of input JSON in the upload or add content request, these parameters are available to update other than system-level fields and the fields that are not considered for update. If provided in modifying the content request, these fields can be ignored and no additional message needs to be shown. Following are the list of parameters:</p> <ul style="list-style-type: none"> • creationDateTime • accessDateTime • parentFolderId • parentFolder • ownerId • ownerName • version, • contentLocationId • revisedDateTime • checkOutTime • contentLocation 	Request body

Sample response

```
{
  "id": "630ddd693cf0cb412311e33c",
  "name": "Banking",
  "contentType": "pdf",
  "comments": "Document related to banking application",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "privilege": "INHERITED",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661858582018,
  "accessDateTime": 1661858582018,
  "version": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
  "dataclass": {
    "dataClassName": "dataclass_banking",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age1": "20"
  },
  "primaryContentId": "630ddd693cf0cb412311e33c",
  "latest": true,
  "checkedOut": false,
  "dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20",
  "accessCount": 0,
  "inTrash": true,
  "indexed": true,
  "nameLowerCase": "banking",
  "docSize": 246
}
```

The following are the description of the response parameters of the *Modify content properties* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.

Parameter	Description
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.

Parameter	Description
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Content checked out

The *Content checked out* API allows you to check out the specific version of a document, once the version is updated.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/checkout/{id}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Content checked out* API, pass the value of the *id* parameters in the request URL. For example, if you want to check out content whose ID is 630e6614dfd49e5ebe551675, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/checkout/630e6614dfd49e5ebe551675

The following are the description of the request parameters of the *Content checked out* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the content.	URL

Sample response

```
{
  "checkedOut": true,
  "primaryContentId": "630dce060fa8b04c913f1f3e",
  "metadata": {
    "key": "v"
  },
  "flag": "COMMITTED",
  "documentType": "png",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "dataclass": {
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "dataClassName": "dataclass_banking",
    "age": "20"
  },
  "privilege": "INHERITED",
  "documentSize": "246",
  "accessDateTime": "Tue Aug 30 08:55:06 GMT 2022",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "ownerName": "John",
  "noOfPages": "1",
  "id": "630e6614dfd49e5ebe551675",
```

```

"contentType": "pdf",
"contentLocationId": "630dce050fa8b04c913f1f3d",
"creationDateTime": "Tue Aug 30 08:44:54 GMT 2022",
"latest": true,
"securityClass": [
  "OWNER"
],
"accessCount": 0,
"comments": "Banking documents",
"indexed": false,
"nameLowerCase": "banking",
"dataClassText": "dataclass_banking 62d7f3c3c0ea8e46159b1040 20",
"checkedOutBy": "62d63eb7016b20476a1c6ae4",
"docSize": 246,
"inTrash": false,
"version": "1.0",
"tenantId": "62d63eb5a2cf20503b23c2ce",
"name": "Banking",
"revisedDateTime": "Tue Aug 30 08:55:06 GMT 2022"
}

```

The following are the description of the response parameters of *Content checked out* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.

Parameter	Description
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.

Parameter	Description
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Setting content version as latest

The *Set content version as latest* API allows you to set the latest version of a required document.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/setlatest/{id}/{version}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Set content version as latest* API, pass the value of the *id* and *version* parameters in the request URL. For example, if you want to set the latest version 1.0 of a document whose ID is 62d63ec46367977f6d813e39, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/setlatest/630ddd693cf0cb412311e33c/1.1

The following are the description of the request parameters of the *Set content version as latest* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the content.	URL
version	Mandatory				Big Decimal	Version of the content.	URL

Sample response

```
{
  "id": "630ddd693cf0cb412311e33c",
  "usedFor": null,
  "name": "Banking",
  "contentType": "pdf",
  "comments": "j",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "securityClass": [
    "OWNER"
  ],
  "ownerName": "John",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630ddd693cf0cb412311e33b",
  "creationDateTime": 1661853033916,
  "revisedDateTime": 1661853033945,
  "accessDateTime": 1661853033945,
  "version": 1,
  "previousVersion": null,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "png",
  "documentSize": "246",
  "metadata": {
    "key": "v"
  },
  "dataclass": {
    "dataClassName": "dataclass_banking",
    "dataClassId": "62d7f3c3c0ea8e46159b1040",
    "age1": "20"
  },
  "primaryContentId": "630ddd693cf0cb412311e33c",
  "latest": true,
  "versionComments": null,
  "checkedOut": false,
  "checkedOutBy": null,
  "lastCheckedInBy": null,
  "lastCheckedInbyUserName": null,
  "previousVersionContentId": null,
  "checkedOutTime": null,
  "parentHierarchy": null,
  "docSize": 246
}
```

The following are the description of the response parameters of the *Set content version as latest* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.

Parameter	Description
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.

Parameter	Description
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Content undo checked out

The *Content undo checked out* API allows you to postpone or undo the changes for a current version of a checked out document.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/contentservice/contents/undocheckout/{id}*
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Content undo checked out* API, pass the value of the *id* parameters in the request URL. For example, if you want to undo the changes for the checked out content whose ID is 630dd9103cf0cb412311e33a, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/contentservice/contents/checkout/630dd9103cf0cb412311e33a

The following are the description of the request parameters of the *Content undo checked out* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the content.	URL

Sample response

```
{
  "id": "630dd9103cf0cb412311e33a",
  "name": "Aug_bill",
  "contentType": "pdf",
  "comments": "August month bill",
  "parentFolderId": "62d63ec46367977f6d813e3c",
  "ownerName": "df",
  "ownerId": "62d63eb7016b20476a1c6ae4",
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "contentLocationId": "630dd9103cf0cb412311e339",
  "privilege": "INHERITED",
  "creationDateTime": 1661849094001,
  "revisedDateTime": 1661858935893,
  "accessDateTime": 1661858935893,
  "version": 1.1,
  "previousVersion": 1,
  "flag": "COMMITTED",
  "noOfPages": "1",
  "documentType": "pdf",
  "documentSize": "246",
  "primaryContentId": "630dce060fa8b04c913f1f3e",
```

```

"latest": true,
"checkedOut": false,
"lastCheckedInBy": "62d63eb7016b20476a1c6ae4",
"previousVersionContentId": "630dce060fa8b04c913f1f3e",
"accessCount": 0,
"inTrash": false,
"indexed": false,
"nameLowerCase": "aug_bill",
"docSize": 246
}

```

The following are the description of the response parameters of *Content undo checked out* API:

Parameter	Description
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last accessed.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the content was created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the content was last modified.
deleted	It indicates whether the content is deleted or not. The true value of the delete parameter defines that the content has been deleted and the clean-up service will remove it at the scheduled interval.
name	It indicates the name of the content. It is a mandatory field to create content.
contentType	It provides the type of content. For example, PDF, Image, and so on.
id	It indicates the content ID.
inTrash	It indicates whether the content or folder is in the trash or not.
privilege	The privilege of the user who is accessing the content ['INHERITED', 'PRIVATE', 'SHARED']. This parameter is mandatory if the content is created as private or shared.
indexed	Indicates whether the content or folder is indexed in search for full-text search API.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the content was last modified.

Parameter	Description
metadata	It is a JSON key-value pair of metadata associated with the content. It is a mandatory parameter while using API to add metadata to content.
ownerId	ID of the user who uploads the content.
ownerName	Name of the user who uploads the content.
parentFolderName	The name of the parent folder.
parentFolderId	It is a system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentHierarchy	Provides folder hierarchy of the content or folders from the root folder.
dataClass	A JSON key-value pair of the data indexing associated with the content.
contentLocationId	ID of a location where content is stored.
comments	A brief description of the content or folder.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
uploadIn	This parameter determines the document's categorization. All documents categorized as "Asset" are available for indexing in the search service and can be searched using search service APIs.
usedFor	Describe the purpose of the content.
primaryContentId	Contains the ID of the base version of the content.
lastCheckedInBy	ID of the user who checked out the document last time.
docSize	The size of a document.
noOfPages	The number of pages in a document.
documentType	It indicates the extension type of a file or document such as png, jpg, pdf, and so on.
documentSize	It indicates the size of the document.
lastCheckedInbyUserName	Name of the user that checked out the document last time.

Parameter	Description
dataClassText	It indicates the combined values of the data class fields.
checkedOut	It indicates whether the document is checked out or not.
checkedOutBy	Name of the user who has checked out the document.
checkedOutTime	It indicates the checked out time of the document.
flag	It indicates the upload status of the content on the Cloud platform such as Azure.
latest	It indicates the latest version of a document based on its version.
previousVersion	It indicates the version of the previous document.
previousVersionContentId	Content ID of the previous document version.
versionComments	Comments to describe the current version of the document.
storageUsed	It displays the used storage.
storageRemaining	It displays the remaining storage.
storageAvailable	It displays the available storage.
version	It is a system-operated field for versioning.
securityClass	The securityClass field is an array to capture the tags applied on a document. A combination of global tag and securityClass is defined to have rights. This combination is assigned to a role. A role is associated with a user. This flow gives user rights that are defined for this combination on the folders and documents on which this combination matches with applied tags in the security class field.

Error codes

The following table describes the error codes and their description related to content services:

Error Code	Error Message
15001	Blank file. Select a file to upload.
15002	Error uploading stored content by using storageCredentialID.
15003	Error storing file in temp folder.

Error Code	Error Message
15004	No metadata associated with the requested content.
15005	No dataclass associated with the requested content.
15006	Invalid content versioning parameter in the request.
15007	Content not checked out.
15008	Content not checked out by user sent in request.
15009	Content already checked out.
150010	Latest content version in use already.
150011	Content is checked out.
150012	Trash folder not found.
150013	Content not in trash folder.
150014	Failed to delete storage location.
150015	File upload limit exceeded.
150016	Invalid Parent folder ID or Parent folder ID is missing.
150017	Invalid Folder ID or Folder ID is missing.
150018	Invalid Source content ID or Source content ID is missing.
150019	Invalid Target folder ID or Target folder ID is missing.
150020	Required parameter is missing.
150022	Invalid Page Number or Page Number must be greater than zero.
150023	Unable to check-in meta content.
150024	Content not found due to invalid primary Content ID or Tenant ID.
150025	Invalid Content ID or Content ID is missing.
150026	Content upload supports uploading of a single file only.

Secret key services

The secret key services are used to generate the integration keys for the app service.

The secret key services include the following APIs:

- [Getting a secret key](#)
- [Getting multiple secret keys](#)
- [Getting all secret keys](#)
- [Validating secret key](#)

Headers

The below table lists the header parameters for secret key services:

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.
userId	Conditional		String	This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed. Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.
appld	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Getting a secret key

The *Get a secret key* API allows you to get a secret key using its ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/secretkeyservice/secretkey/{id}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get a secret key* API, pass the value of the *id* parameter in the request URL. For example, if you want to get a secret key whose ID is `6321cc306c23f14b74e65108`, then pass the request URL as follows:
`https://<IP address>:<Port>/ecmapi/secretkeyservice/secretkey/6321cc306c23f14b74e65108`

The following table lists the descriptions of request parameters of the *Get a secret key* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	It indicates the secret key ID which you want to get.	URL

Sample response

```
{
  "id": "6321cc306c23f14b74e65108",
  "tenantId": "6321cbfe601d561c424f89bc",
  "secretKeyInfo": [
    {
      "ownerId": "System",
      "key": "K2xKRmUvc3dkUFVNTXJYQW5EakFTN1NjZ1pUekVCTW12RDV2Ly95RjIvblZaZnFsV0dLbDZKbEltbWRPThZVw",
      "roleIds": [
        "632h4w306c23f14b74gqo6f3"
      ],
      "creationDateTime": "2022-09-14T12:54:21.858Z",
      "expiryDateTime": "2023-09-14T12:54:21.858Z"
    }
  ]
}
```

The following table lists the descriptions of response parameters of the *Get a secret key* API:

Parameter	Description
id	ID of the secret key.
tenantId	ID of the registered tenant.
secretKeyInfo	It contains the key-value pairs which provide the required information of the secret key.
ownerId	ID of the user who generates the secret key.
key	It indicates the generated secret key.
roleIds	List of the role IDs.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was generated.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was last modified.

Getting multiple secret keys

The *Get multiple secret keys* API allows you to get multiple secret keys data as required.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/secretkeyservice/secretkey/multiplekey`
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "key": "bjc3MDluam1od0JvZUJJbWRoYS9xNmxlMWp3c3lySkpMZVdXQzdVdK3ZwNjJnRz1
WKyttVct3bS9Xc1pwd0tQZw",
  "key": "TVpjaWkxbGc5YmFEU3psaEZWOURTMkdYWj1nc1d6emNZcmZVYVFxd2o5OU1XMUt
xbVArdeEya2d1dVAvdHNkRQ"
}
```

The following table lists the descriptions of request parameters of the *Get multiple secret keys* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
keys	Optional				ArrayList<String>	List of multiple secret keys. that you want to find.	Request body
sortOn	Optional	creationDateTime, expiryDateTime			String	It is used to sort the secret key.	URL
sortOrder	Optional	ASC-Ascending DESC-Descending			String	It is used to define the sorting order of the secret key.	URL

Sample response

```
{
  {
    "id": "6321cc306c23f14b74e65108",
    "tenantId": "6321cbfe601d561c424f89bc",
    "secretKeyInfo": [
      {
        "ownerId": "System",
        "key": "bjc3MDluam1od0JvZUJbWRoYS9xNmxlMWp3c3lySkpMZVd
XQzdVzKzZwNjJnRzlwKyttVct3bS9Xclpwd0tQZw",
        "roleIds": [
          "632h4w306c23f14b74gqo6f3"
        ],
        "creationDateTime": "2022-09-14T12:54:21.858Z",
        "expiryDateTime": "2023-10-14T12:54:21.858Z"
      },
      {
        "ownerId": "System",
        "key": "TVpjaWkxbGc5YmFEU3psaEZWOURTMkdYWjlnclD6emNZcm
ZVYVFxd2o5OU1XMUtXbVArdeEya2dldVAvdHNkRQ",
        "creationDateTime": "2022-09-14T12:42:24.239Z",
        "expiryDateTime": "2023-09-14T12:42:24.239Z"
      }
    ]
  }
}
```

The following table lists the descriptions of response parameters of the *Get multiple secret keys* API:

Parameter	Description
id	ID of the secret key.
tenantId	ID of the registered tenant.
secretKeyInfo	It contains the key-value pairs which provide the required information of the secret key.
ownerId	ID of the user who generates the secret key.
key	It indicates the generated secret key.
roleIds	List of the role IDs.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was generated.

Parameter	Description
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was last modified.

Getting all secret keys

The *Get all secret keys* API allows you to get all existing secret keys using tenant ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/secretkeyservice/secretkey?sortOn=ownerId,sortOrder=ASC`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

The following table lists the descriptions of request parameters of the *Get all secret keys* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
sortOn	Optional	creationDateTime, expiryDateTime			String	It is used to sort the secret key.	Query
sortOrder	Optional	ASC - Ascending DESC - Descending			String	It is used to define the sorting order of the secret key.	Query

Sample response

```
{
  "id": "6321cc306c23f14b74e65108",
  "tenantId": "6321cbfe601d561c424f89bc",
  "secretKeyInfo": [
    {
      "ownerId": "System",
      "key": "bjc3MDluam1od0JvZUJJbWRoYS9xNmxlMWp3c3lySkpMZVdXQzdV
KzZwNjJnRz1WKyttVCt3bS9Xclpwd0tQZw",
      "roleIds": [
        "632h4w306c23f14b74gqo6f3"
      ],
      "creationDateTime": "2022-09-14T12:54:21.858Z",
      "expiryDateTime": "2023-10-14T12:54:21.858Z"
    },
    {
      "ownerId": "System",
      "key": "TVpjaWkxbGc5YmFEU3psaEZWOURTMkdYWjlncl1d6emNZcmZVYVFX
d2o5OU1XMUtxbVArdEEya2dldVAvdHNkRQ",
      "creationDateTime": "2022-09-14T12:42:24.239Z",
      "expiryDateTime": "2023-09-14T12:42:24.239Z"
    }
  ]
}
```

The following table lists the descriptions of response parameters of the *Get all secret keys* API:

Parameter	Description
id	ID of the secret key.
tenantId	ID of the registered tenant.
secretKeyInfo	It contains the key-value pairs which provide the required information of the secret key.
ownerId	ID of the user who generates the secret key.
key	It indicates the generated secret key.
roleIds	List of the role IDs.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was generated.

Parameter	Description
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was last modified.

Validating secret key

The *Validate secret key* API allows you to validate the secret key data.

- **Service URL:** *https://<IP address>:<Port>/ecmapi/secretkeyservice/secretkey/key*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "key": "bjc3MDluam1od0JvZUJJbWRoYS9xNmxlMWp3c3lYSkpMZVdXQzdVZwNjZlWkYttVct3bS9Xclpwd0tQZw"
}
```

The following table lists the descriptions of request parameters of the *Validate secret key* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
key	Mandatory				String	It indicates the secret Key which you want to validate.	Request body

Sample response

```
{
  "id": "6321cc306c23f14b74e65108",
  "tenantId": "6321cbfe601d561c424f89bc",
}
```

```

"ownerId": "System",
"key": "bjc3MDluam1od0JvZUJbWRoYS9xNmxlMWp3c3lySkpMZVdXQzdvKzZwNjJnRz1WKyttVct3bS9Xclpwd0tQZw",
"roleIds": [
  "632h4w306c23f14b74gqo6f3"
],
"creationDateTime": "2022-09-14T12:54:21.858Z",
"expiryDateTime": "2023-10-14T12:54:21.858Z"
}

```

The following table lists the descriptions of response parameters of the *Validate secret key* API:

Parameter	Description
id	ID of the secret key.
tenantId	ID of the registered tenant.
secretKeyInfo	It contains the key-value pairs which provide the required information of the secret key.
ownerId	ID of the user who generates the secret key.
key	It indicates the generated secret key.
roleIds	List of the role IDs.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was generated.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the secret key was last modified.

Error codes

The below table lists the error codes for secret key services:

Error code	Error message
61001	Secret key creation error.
61002	Secret key is not found.
61003	Roles addition error.
61004	Roles are not found.

Error code	Error message
61005	Secret key updation error.
61006	Secret key does not exist.
61007	Secret key is expired.
61008	Invalid Sort On value.
61009	Invalid Sort Order value.
610010	Secret key is required.

Security classification services

Security classification service allows you to segregate information available to users of an organization. You can create security classes and global tags and associate them with documents and folders. It is also possible to associate the roles with security classes and global tags and define the access rights to the system users accordingly.

For example, a user in the customer service department is not allowed to access the sales repository or vice-versa. Similarly, permission to access confidential data is allowed to a specific set of users. Any user of the system can access the content marked as public.

Prerequisites:

- A tenant ID is required to access the NewgenONE Content Cloud services. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
- Parameter org is required, its value is *ECM* to try out any API.
- Security classes and Global tags need to be pre-defined to associate with the folders and documents. By default, security classes such as Top Secret, Secret, Confidential, and Non-Confidential are available to use.

The security classification services include the following APIs:

- [Getting list of all security classes](#)
- [Getting list of all global tags](#)

Headers

The below table lists the header parameters for security classification services:

Parameters	Mandatory/Optional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameters	Mandatory/ Optional	Accepted Values	Data Type	Description
userId	Conditional		String	This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed. Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.
appId	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Getting list of all security classes

The *Get list of all security classes* API allows you to get all existing security classes using tenant ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/classificationservice/securityClass`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample response

```
{
  [
    {
      "id": "622f08b9cf2837746ea4730a",
      "securityClass": "Home Loan",
      "tenantId": "621c503c10c9f42a1408ad2b",
      "ownerId": "621c503fa2c0ae38fe7aaa64",
      "description": "Security class created for home loan",
      "creationDateTime": "2022-03-14T09:19:53.959Z"
    },
    {
      "id": "622f08b9cf2837746ea4730a",
      "securityClass": "Car Loan",
      "tenantId": "621c503c10c9f42a1408ad2b",
      "ownerId": "621c503fa2c0ae38fe7aaa64",
      "description": "Security class created for car loan",
      "creationDateTime": "2022-03-14T09:19:53.959Z"
    }
  ]
}
```

The following table lists the descriptions of response parameters of the *Get list of all security classes* API:

Parameter	Description
id	ID of the security class.

Parameter	Description
SecurityClass	Name of the security class.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the security class.
description	A brief description of security class.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the security class was created.

Getting list of all global tags

The *Get list of all global tags* API allows you to get all existing global tags using tenant ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/classificationsservice/globalTag`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample response

```
{
  [
    {
      "id" : "622f08b9cf2837746ea4730a",
      "globalTag" : "Finance",
      "tenantId" : "621c503c10c9f42a1408ad2b",
      "ownerId" : "621c503fa2c0ae38fe7aaa64",
      "description" : "Tag created to manage financial documents",
      "creationDateTime" : "2022-03-14T09:19:53.959Z"
    },
    {
      "id" : "622f08b9cf2837746ea4730a",
      "globalTag" : "Project Planning",
      "tenantId" : "621c503c10c9f42a1408ad2b",
      "ownerId" : "621c503fa2c0ae38fe7aaa64",
      "description" : "Tag created to manage project planning documents",
      "creationDateTime" : "2022-03-14T09:19:53.959Z"
    }
  ]
}
```

```

    ]
}

```

The following table lists the descriptions of response parameters of the *Get list of all global tags* API:

Parameter	Description
id	ID of the global tag.
globalTag	Name of the global tag.
tenantId	ID of the registered tenant.
ownerId	ID of the user who creates the global tag.
description	A brief description for the global tag.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the global tag was created.

Error codes

The below table lists the error codes for security classification services:

Error code	Error message
63001	Security Classification creation error.
63002	Security Classification deletion error.
63003	Security Classification is not found.
63004	Global Tag creation error.
63005	Global Tag deletion error.
63006	Global Tag is not found.
63007	Duplicate Security Class creation is not allowed.
63008	Duplicate Global Tag creation is not allowed.

Data class services

The data class is a set of indexes that can be associated with any document or folder by providing a unique entity to them. These indexes store the values provided so that the user can perform a search on them. The indexes or fields of a data class data can be of different types such as Double, Long, Date, and Boolean.

When a data class is associated with a document or a folder it becomes an integral part of the folder or document property. It can be assigned, modified, and searched upon. For example, if a user creates a data class named employee, containing fields like emp_id, emp_name, and emp_DOB, it can be associated with all the employees' records and provides a quick search on any fields. Users can also provide a constraint on fields such as mandatory, optional, and secure.

The data class services include the following APIs:

- [Get data class by ID](#)
- [Get data class by name](#)
- [Search data class](#)

Headers

The below table lists the header parameters for data indexing services:

Parameter	Mandatory/Optional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.

Parameter	Mandatory/ Optional	Accepted Values	Data Type	Description
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.
userId	Conditional		String	This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed. Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.
appId	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Getting data class by ID

The *Get data class by ID* API allows you to get the data class using its ID that is assigned while creating the data class.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/dataclassservice/dataclass/{Id}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get data class by ID* API, pass the value of the *id* parameter in the request URL. For example, if you want to get a data class whose ID is *62d63ec46367977f6d813e3c*, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/dataclassservice/dataclass/62d63ec46367977f6d813e3c

The following is the description of the request parameter of the *Get data class by ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory				String	ID of the data class.	URL

Sample response

```
{
  "id": "62d63ec46367977f6d813e3c",
  "dataClassName": "Banking",
  "comments": "Data Class for account creation",
  "dataFields": [
    {
      "fieldIndex": "014c948e-72be-4511-84a1-05575ba18b05",
      "fieldName": "Full_Name",

```

```

    "fieldType": "LONG",
    "fieldOrder": "0",
    "isFieldMandatory": "false",
    "isFieldSecure": "false"
  },
  {
    "fieldIndex": "54580e22-edd5-4514-8f18-1fcc85a2484c",
    "fieldName": "Address",
    "fieldType": "STRING",
    "fieldOrder": "0",
    "isFieldMandatory": "false",
    "isFieldSecure": "false"
  },
  {
    "fieldIndex": "8877f850-4925-4938-a85f-8e1d9c849955",
    "fieldName": "Occupation",
    "fieldType": "STRING",
    "fieldOrder": "0",
    "isFieldMandatory": "false",
    "isFieldSecure": "false"
  },
  {
    "fieldIndex": "495a8ed2-95e8-491a-83d4-e078a83613cc",
    "fieldName": "Age",
    "fieldType": "LONG",
    "fieldDescription": "This is a client age.",
    "fieldOrder": "20",
    "isFieldMandatory": "false",
    "isFieldSecure": "false"
  }
],
"fieldCount": 4,
"tenantId": "62d63eb5a2cf20503b23c2ce",
"revisedDateTime": "2022-09-19T06:21:12.784Z",
"accessDateTime": "2022-09-19T06:21:12.784Z",
"lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
}

```

The following are the description of the response parameters of the *Get data class by ID* API:

Parameter	Description
id	ID of the data class.
dataClassName	Name of the data class. This field cannot be duplicated for the same tenant.
comments	Description of the data class.

Parameter	Description
dataFields	Data Fields of the data class.
fieldIndex	It is a system-generated ID corresponding to each field name of the data class.
fieldName	Name of the data field. This field cannot be duplicated in the same data class.
fieldType	Type of the data field.
fieldDescription	Description of the data field.
fieldOrder	Order of the data field. This field has no validation other than it accepts positive numbers. It is the responsibility of the end-user or developer to provide data according to the use case.
isFieldMandatory	This field describes whether the data field is mandatory or not.
isFieldSecure	This field describes whether the data field is secured or not. Secured data class field values will be stored as encrypted values. Based on the rights of the user to view secured data, the decrypted value appears. The value of this field cannot be changed after the creation of data field in the application.
fieldCount	It indicates the total count of data class fields passed in the request body.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the data class was last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the data class was last accessed.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the data class was last modified.

Getting data class by name

The *Get data class by name* API allows you to get the data class using its name that is assigned while creating the data class.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/dataclassservice/dataclass/{dataClassName}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get data class by name* API, pass the value of the *dataClassName* parameter in the request URL. For example, if you want to get a data class whose name is Banking, then pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/dataclassservice/dataclass/Banking

The following is the description of the request parameter of the *Get data class by name* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
dataClassName	Mandatory				String	Name of the data class.	URL

Sample response

```
{
  "id": "63280a583c122d6308550cd3",
  "dataClassName": "Banking",
  "comments": "Data class for account creation.",
  "dataFields": [
    {
      "fieldIndex": "014c948e-72be-4511-84a1-05575ba18b05",
      "fieldName": "Name",
```

```

        "fieldType": "LONG",
        "fieldMaxLength": "20",
        "fieldOrder": "0",
        "isFieldMandatory": "false",
        "isFieldSecure": "false"
    },
    {
        "fieldIndex": "495a8ed2-95e8-491a-83d4-e078a83613cc",
        "fieldName": "Age",
        "fieldType": "LONG",
        "fieldDescription": "Age of the customer",
        "fieldMaxLength": "20",
        "fieldOrder": "20",
        "isFieldMandatory": "false",
        "isFieldSecure": "false"
    }
],
"fieldCount": 2,
"tenantId": "62d63eb5a2cf20503b23c2ce",
"revisedDateTime": "2022-09-19T06:21:12.784Z",
"accessDateTime": "2022-09-19T06:21:12.784Z",
"lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
}

```

The following are the description of the response parameters of the *Get data class by name* API:

Parameter	Description
id	ID of the data class.
dataClassName	Name of the data class. This field cannot be duplicated for the same tenant.
comments	Description of the data class.
dataFields	Data Fields of the data class.
fieldIndex	It is a system-generated ID corresponding to each field name of the data class.
fieldName	Name of the data field. This field cannot be duplicated in the same data class.
fieldType	Type of the data field.
fieldDescription	Description of the data field.

Parameter	Description
fieldOrder	Order of the data field. This field has no validation other than it accepts positive numbers. It is the responsibility of the end-user or developer to provide data according to the use case.
isFieldMandatory	This field describes whether the data field is mandatory or not.
isFieldSecure	This field describes whether the data field is secured or not. Secured data class field values will be stored as encrypted values. Based on the rights of the user to view secured data, the decrypted value appears. The value of this field cannot be changed after the creation of data field in the application.
fieldCount	It indicates the total count of data class fields passed in the request body.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the data class was last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the data class was last accessed.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the data class was last modified.

Searching data class

The *Searching data class* API allows you to search a data class using its name that is assigned while creating the data class.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/dataclassservice/dataclass/dataClassName={dataclass name}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Searching data class* API, pass the value of the *dataClassName* parameter in the query parameter of the request URL. For example, if you want to search a data class whose name is Banking, then pass the request URL as follows:
*http or https://<IP address>:<Port>/ecmapi/dataclassservice/dataclass/
 dataClassName=Banking*

The following is the description of the request parameter of the *Searching data class* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
dataClassName	Mandatory				String	Name of the data class.	Query parameter

Sample response

```
{
  "id": "63280a583c122d6308550cd3",
  "dataClassName": "Banking",
  "comments": "Data class for account creation.",
  "dataFields": [
    {
      "fieldIndex": "014c948e-72be-4511-84a1-05575ba18b05",
      "fieldName": "Name",
      "fieldType": "LONG",
      "fieldMaxLength": "20",
      "fieldOrder": "0",
      "isFieldMandatory": "false",
      "isFieldSecure": "false"
    },
    {
      "fieldIndex": "495a8ed2-95e8-491a-83d4-e078a83613cc",
      "fieldName": "Age",
      "fieldType": "LONG",
      "fieldDescription": "Age of the customer",
      "fieldMaxLength": "20",
      "fieldOrder": "20",
      "isFieldMandatory": "false",
      "isFieldSecure": "false"
    }
  ]
}
```

```

    }
  ],
  "fieldCount": 2,
  "tenantId": "62d63eb5a2cf20503b23c2ce",
  "revisedDateTime": "2022-09-19T06:21:12.784Z",
  "accessDateTime": "2022-09-19T06:21:12.784Z",
  "lastModifiedByUserId": "62d63eb7016b20476a1c6ae4"
}

```

The following are the description of the response parameters of the *Searching data class* API:

Parameter	Description
id	ID of the data class.
dataClassName	Name of the data class. This field cannot be duplicated for the same tenant.
comments	Description of the data class.
dataFields	Data Fields of the data class.
fieldIndex	It is a system-generated ID corresponding to each field name of the data class.
fieldName	Name of the data field. This field cannot be duplicated in the same data class.
fieldType	Type of the data field.
fieldDescription	Description of the data field.
fieldOrder	Order of the data field. This field has no validation other than it accepts positive numbers. It is the responsibility of the end-user or developer to provide data according to the use case.
isFieldMandatory	This field describes whether the data field is mandatory or not.
isFieldSecure	This field describes whether the data field is secured or not. Secured data class field values will be stored as encrypted values. Based on the rights of the user to view secured data, the decrypted value appears. The value of this field cannot be changed after the creation of data field in the application.
fieldCount	It indicates the total count of data class fields passed in the request body.

Parameter	Description
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the data class was last modified.
accessDateTime	It is a system-operated field to indicate the signing-in date and time when the data class was last accessed.
lastModifiedByUserId	It is a system-operated field to keep the logs of the user IDs through which the data class was last modified.

Error codes

The following table describes the error codes and their description related to data indexing services:

Error code	Error message
23001	Data class creation error.
23002	Data class is not found.
23003	Data class already exists.
23004	Data class ID is invalid or blank.
23005	Data class name is invalid or blank.
23006	Data class field names cannot be duplicate.
23007	Data class metadata is not as per the data class definition.
23008	Field name is required to add data field.
23009	Field index does not exist.
230010	Field action is not valid. The expected value ADD.
230011	Field action is not valid. The expected value is any of UPDATE or DELETE.
230012	Data class is not exist.

Notes services

Notes service allows you to store certain annotations or remarks on content. It is extra information (such as a document reviewer comment) associated with a particular point in a document or other piece of information. It can be a note that includes a comment or explanation. Thus, notes become a very handy feature in collaborative environments where you can highlight certain portions of the page to draw attention. For example, adding notes to a folder, documents, or any other object for collaboration purposes, pending task management, and so on.

Prerequisites:

- A tenant ID is required to access the NewgenONE Content Cloud services. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.
- Parameter org is required, its value is *ECM* to try out any API.

The notes services include the following APIs:

- [Creating notes](#)
- [Modifying notes by notes ID](#)
- [Getting notes by notes ID](#)
- [Getting notes by object ID](#)
- [Deleting notes](#)

Headers

The below table lists the header parameters for notes services:

Parameter	Mandatory/ Conditional	Accepted Value	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/ json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	The accessToken is an authentication token. It is obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameter	Mandatory/ Conditional	Accepted Value	Data Type	Description
userId	Conditional		String	This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web to Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed. Since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario, this parameter must be sent. Otherwise, services may raise an exception.
appId	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Creating notes

The *Create notes* API allows you to create notes.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/notesservice/notes/`
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
{
  "ownerName": "James",
  "objectId": "61dfe968cec73a77af3d75c8",
  "objectType": "Media",
  "comment": "Update the video KYC"
}
```

The following table lists the descriptions of request parameters of the *Create notes* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
comment	Mandatory				String	It is a mandatory parameter while creating the note. It accepts alphanumeric, comma, full-stop, and spaces.	Request body
objectId	Optional				String	ID of the object. It accepts only the alphanumeric characters.	Request body

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
ownerName	Mandatory				String	Name of the user who is creating the notes. It accepts alphanumeric characters along with spaces, underscores, and hyphens. The first character is always an alphabet.	Request body
objectType	Optional	Example: Folder, Document, and Media.			String	Type of the object. It accepts only alphabets.	Request body

Sample response

```
{
  {
    "id" : "620f72c9f8561d5217bc2283",
    "ownerId" : "61ad989d39f23d0523b753dc",
    "tenantId" : "61ad989c148e17379fccfa77",
    "ownerName" : "James",
    "objectId" : "61dfe968cec73a77af3d75c8",
    "objectType" : "Media",
    "comment" : "Update the video KYC",
    "type" : "N",
    "creationDateTime" : "2022-02-18T10:19:53.744Z",
    "revisedDateTime" : "2022-02-18T10:19:53.744Z"
  }
}
```

The following table lists the descriptions of response parameters of the *Create notes* API:

Parameter	Description
id	ID of the notes.
ownerId	ID of the user who creates the notes.
tenantId	ID of the registered tenant.
ownerName	Name of the user who creates the notes.
objectId	ID of the object.
objectType	Type of the object.
comment	It provides a brief description for creating the notes.
type	It indicates the type of notes. The value of this parameter is set as 'N'.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were last modified.

Getting notes by notes ID

The *Get notes by notes ID* API allows you to get the existing notes using notes ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/notesservice/notes/{{id}}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get notes by notes ID* API, pass the value of the *id* parameters in the request URL. For example, if you want to get the notes whose ID is *620f72c9f8561d5217bc2283*, then pass the request URL as follows:

https://<IP address>:<Port>/ecmapi/notesservice/notes/620f72c9f8561d5217bc2283

```
{
  "ownerName": "James",
  "comment": "Update the video KYC"
}
```

The following table lists the descriptions of request parameters of the *Get notes by notes ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	ID of the notes to get the existing notes. It accepts only alphanumeric characters.	URL

Sample response

```
{
  {
    "id" : "620f72c9f8561d5217bc2283",
    "ownerId" : "61ad989d39f23d0523b753dc",
    "tenantId" : "61ad989c148e17379fccfa77",
    "ownerName" : "James",
    "objectId" : "61dfe968cec73a77af3d75c8",
    "objectType" : "Media",
    "comment" : "Update the video KYC",
    "type" : "N",
    "creationDateTime" : "2022-02-18T10:19:53.744Z",
```



```

    "revisedDateTime" : "2022-02-18T10:19:53.744Z"
  }
}

```

The following table lists the descriptions of response parameters of the *Get notes by notes ID* API:

Parameter	Description
id	ID of the notes.
ownerId	ID of the user who creates the notes.
tenantId	ID of the registered tenant.
ownerName	Name of the user who creates the notes.
objectId	ID of the object.
objectType	Type of the object.
comment	It provides a brief description for creating the notes.
type	It indicates the type of notes. The value of this parameter is set as 'N'.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were last modified.

Getting notes by object ID

The *Get notes by object ID* API allows you to get all existing notes using tenant ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/notesservice/notes/?objectId={{id}}`
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Get notes by object ID* API, pass the value of the *objectId* query parameter in the request URL. For example, if you want to get the notes whose object ID is `620f72c9f8561d5217bc2283`, then pass the request URL as follows:

```
https://<IP address>:<Port>/ecmapi/notesservice/notes/?
objectId=620f72c9f8561d5217bc2283
```

The following table lists the descriptions of request parameters of the *Get notes by object ID* API:

Parameter	Mandatory/Optional	Accepted Value	Default Value	System Field	Data Type	Description	Parameter Type
objectId	Mandatory				String	Object ID of the notes. It accepts only alphanumeric characters.	Query Param

Sample response

```
{
  [
    {
      "id" : "620f72c9f8561d5217bc2283",
      "ownerId" : "61ad989d39f23d0523b753dc",
      "tenantId" : "61ad989c148e17379fccfa77",
      "ownerName" : "James",
      "objectId" : "61dfe968cec73a77af3d75c8",
      "objectType" : "Media",
      "comment" : "Update the video KYC",
      "type" : "N",
      "creationDateTime" : "2022-02-18T10:19:53.744Z",
      "revisedDateTime" : "2022-03-11T10:19:53.744Z"
    },
    {
      "id" : "620f72c9f8561d5217bc2283",
      "ownerId" : "61ad989d39f23d0523b753dc",
      "tenantId" : "61ad989c148e17379fccfa77",
      "ownerName" : "Roy",
      "objectId" : "61dfe968cec73a77af3d75c8",
```

```

    "objectType" : "Media",
    "comment" : "Verify the customer documents",
    "type" : "N",
    "creationDateTime" : "2022-02-18T10:19:53.744Z",
    "revisedDateTime" : "2022-03-11T10:19:53.744Z"
  }
]
}

```

The following table lists the descriptions of response parameters of the *Get notes by object ID* API:

Parameter	Description
id	ID of the notes.
ownerId	ID of the user who creates the notes.
tenantId	ID of the registered tenant.
ownerName	Name of the user who creates the notes.
objectId	ID of the object.
objectType	Type of the object.
comment	It provides a brief description for creating the notes.
type	It indicates the type of notes. The value of this parameter is set as 'N'.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were last modified.

Modifying notes by notes ID

The *Modify notes by notes ID* API allows you to update the notes data using notes ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/notesservice/notes/{{id}}`
- **Method type:** PUT
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Modify notes by notes ID* API, pass the value of the *id* parameters in the request URL. For example, if you want to modify the notes whose ID is *620f72c9f8561d5217bc2283*, then pass the request URL as follows:

https://<IP address>:<Port>/ecmapi/notesservice/notes/620f72c9f8561d5217bc2283

```
{
  "ownerName": "James",
  "comment": "Update the video KYC"
}
```

The following table lists the descriptions of request parameters of the *Modify notes by notes ID* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Value
id	Mandatory			Yes	String	ID of the notes. It is a mandatory parameter while modifying the comment or owner name. It accepts only alphanumeric characters.	URL
comment	Optional				String	It provides a description of the notes. It accepts alphanumeric, comma, full-stop, and spaces.	Request body

Parameter	Mandatory/ Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Value
ownerName	Optional				String	Name of the user who is creating the notes. It accepts alphanumeric characters along with spaces, underscores, and hyphens. The first character is always an alphabet.	Request body

Sample response

```
{
  {
    "id" : "620f72c9f8561d5217bc2283",
    "ownerId" : "61ad989d39f23d0523b753dc",
    "tenantId" : "61ad989c148e17379fccfa77",
    "ownerName" : "James",
    "objectId" : "61dfe968cec73a77af3d75c8",
    "objectType" : "Media",
    "comment" : "Update the video KYC",
    "type" : "N",
    "creationDateTime" : "2022-02-18T10:19:53.744Z",
    "revisedDateTime" : "2022-02-18T10:19:53.744Z"
  }
}
```

The following table lists the descriptions of response parameters of the *Modify notes by notes ID* API:

Parameter	Description
id	ID of the notes.
ownerId	ID of the user who creates the notes.

Parameter	Description
tenantId	ID of the registered tenant.
ownerName	Name of the user who creates the notes.
objectId	ID of the object.
objectType	Type of the object.
comment	It provides a brief description for creating the notes.
type	It indicates the type of notes. The value of this parameter is set as 'N'.
creationDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were created.
revisedDateTime	It is a system-operated field to indicate the signing-in date and time when the notes were last modified.

Deleting notes

The *Delete notes* API allows you to delete an existing role from the system using the role ID.

- **Service URL:** `https://<IP address>:<Port>/ecmapi/notesservice/notes/{{id}}`
- **Method type:** DELETE
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Delete notes* API, pass the value of the *id* parameter in the request URL. For example, if you want to delete the notes whose ID is `62bc4ef9e7c42f246f4ca46a`, then pass the request URL as follows:
`https://<IP address>:<Port>/ecmapi/notesservice/notes/62bc4ef9e7c42f246f4ca46a`

The following is the description of the request parameter of the *Delete notes* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory			Yes	String	It indicates the notes ID which you want to delete from the system. It accepts only alphanumeric characters.	URL
ownerId	Mandatory				string	It indicates the owner ID of the user who creates the notes. It accepts only alphanumeric characters.	Header

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Error codes

The below table lists the error codes for notes services:

Error code	Error message
88001	Notes creation error.
88002	Notes deletion error.

Error code	Error message
88003	Notes not found.
88004	Nothing to update or modify, all mandatory fields are required.

Search service

NewgenONE Content Cloud Search Service provides indexing and querying capabilities for the data uploaded to NewgenONE Content Cloud storage. It acts as a service framework that is intended to provide developers with powerful data discovery capabilities like facets-based full-text search (FTS) on files and folder names with advanced filtering based on comments, owner name, document type, keywords, and data class (metadata).

The search service includes the [Searching files and folders](#) API.

Headers

The below table lists the header parameters for annotation services:

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, a tenant ID is generated. You can copy the tenant ID from the activation email you received while signing in to the NewgenONE Content Cloud platform.

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	An accessToken is an authentication token obtained from the login API.
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
userId	Conditional		String	<p>This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web-to-Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed.</p> <p>This parameter must be sent since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario. Otherwise, services may raise an exception.</p>
appId	Mandatory		String	<p>This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.</p>

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Searching files and folders

The *Easy Search API* allows you to search files and folders based on the search query fields.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/searchservice/search/easysearch*
- **Method type:** POST
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

```
"params":
{
  "assetType": [
    "fts",
    "folder",
    "document"
  ],
  "searchString": [
    "policy*"
  ],
  "owner": [
    "5f2c60e745ea4e0005554fe9"
  ],
  "documentType": [
    "PDF", "PNG"
  ],
  "keyword": ["kyc"],
```

```
"dataclass":["alpha"]
}
```

The following table lists the descriptions of request parameters of the *Easy Search API*:

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
params	Mandatory				JSON	It is the key of base JSON that contains all the below fields.	Request body
searchString	Mandatory				Array of Strings	A string based on which you want to execute the search operation.	Request body
owner	Optional				Array of Strings	It indicates a list of owner names based on which you want to execute the search operation.	Request body
documentType	Optional				Array of Strings	The type of document based on which you want to execute the search operation.	Request body

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
dataclass	Optional				JSON Array	The dataclass names of the objects you want to search.	Request body
batch	Optional				JSON Array	The batch size of searched items. The value you provide is considered the batch size.	Request body

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
skip	Optional				JSON Array	It is used for batch selection. For example, if the batch size of the searched items is 10 each and you want to pick the 5th batch. Then, in such a case, set skip to 40 to leave the first four batches and pick the 5th (next) one. The value you provide is considered for skipping items.	Request body

Sample response

```
[
  {
    "accessDateTime": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
    "comments": "string",
    "content": "string",
    "contentLocationId": "string",
    "contentType": "string",
    "creationDateTime": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
    "dataclass": {
```

```

    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "dataclassIndexingText": [
    "string"
  ],
  "doc_id": "string",
  "documentSize": "string",
  "documentType": "string",
  "flag": "string",
  "id": "string",
  "keywords": [
    "string"
  ],
  "latest": true,
  "metadata": {
    "additionalProp1": "string",
    "additionalProp2": "string",
    "additionalProp3": "string"
  },
  "name": "string",
  "noOfPages": "string",
  "ownerId": "string",
  "ownerName": "string",
  "parentFolderId": "string",
  "parentFolderName": "string",
  "primaryContentId": "string",
  "revisedDateTime": "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'",
  "rid": "string",
  "tenantId": "string"
}
]

```

The following table lists the descriptions of request parameters of the *Easy Search API*:

Parameter	Description
accessDateTime	It is a system-operated field that indicates the signing-in date and time when the content was last accessed.
comments	A brief description of the content or folder.
content	It refers to the document content that matches the FTS search string.
contentLocationId	The location ID where the content is stored.

Parameter	Description
contentType	It is a system-operated field. If you are uploading general content in a PDF form, then the content type is 'pdf'. Similarly, if you are uploading content of 'jpg (picture)' type, then the content type is 'png'.
creationDateTime	A system-operated field to indicate the signing-in date and time when the content was created.
dataclass	A JSON key-value pair of the data indexing associated with the content.
documentSize	The size of the document.
documentType	The extension type of a file or document such as png, jpg, pdf, and others.
flag	It indicates the upload status of the content on a Cloud platform such as Azure.
id	The ID of the document.
keyword	A list of keywords of string type associated with documents.
name	The name of the content. It is a mandatory field to create the content.
noOfPages	The number of pages in a document.
ownerId	The ID of the user who uploads the content.
ownerName	The name of the user who uploads the content.
parentFolderId	A system-generated ID of the parent folder where the required folder is existing. It is a mandatory field while creating a folder.
parentFolderName	The name of the parent folder.
primaryContentId	The ID of the base version of the content.
revisedDateTime	A system-operated field to indicate the signing-in date and time when the content was last modified.
tenantId	When a client registers at the NewgenONE Content Cloud platform, the tenant ID is generated. You can copy the tenant ID from the activation email that is obtained while signing up for the NewgenONE Content Cloud platform.

Annotation services

The annotation services allow you to save the annotations in a document. It also allows you to fetch the annotations associated with the documents.

The annotation services include the following APIs:

- [Save annotation](#)
- [Get annotation data](#)

Headers

The below table lists the header parameters for annotation services:

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, a tenant ID is generated. You can copy the tenant ID from the activation email you received while signing in to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	An accessToken is an authentication token obtained from the login API.

Parameter	Mandatory/ Conditional	Accepted Values	Data Type	Description
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.
userId	Conditional		String	<p>This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web-to-Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed.</p> <p>This parameter must be sent since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario. Otherwise, services may raise an exception.</p>

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
appId	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Saving an annotation

The *Save Annotation API* allows you to add and save annotations in a document.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/annotationservice/annotations/saveAnnotation*
- **Method type:** POST
- **Request content type:** APPLICATION_FORM_URL_ENCODED_VALUE
- **Response content type:** TEXT_PLAIN_VALUE

Sample request

While calling the *Save Annotation API*, pass the value of the *id* and *documentId* query parameters in the request URL. For example, if you want to add an annotation in a document with id as 64a3e696e0bc1f048f32900e on the page number having id as 1. Then, pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/annotationservice/annotations/saveAnnotation?id=1&documentId=64a3e696e0bc1f048f32900e

x-www-form-urlencoded:

```

{
  [AnnotationGroupHeader]:
  TotalGroups=1:
  [Group1]:
  Name=fghjhj:
  LoginUserRights=VM:
  AnnotationGroupIndex=0:
  [fghjhjAnnotationHeader]:
  TotalAnnotations=1:
  NoOfLine=1:
  [fghjhjLine1]:
  X1=174:
  Y1=173:
  X2=175:
  Y2=173:
  Color=255:
  TimeOrder=2022,10,31,16,15,01:
  MouseSensitivity=1:
  AnnotationGroupID=fghjhj:
  UserID=fghjhj:
  Rights=VM:
  Thickness=1:
  Style=0:
}

```

The following table lists the descriptions of parameters of the *Save Annotation API*:

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory				String	The document page number where you want to add an annotation.	Query param

Parameter	Mandatory/ Optional/ Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
documentId	Mandatory		1		String	The unique ID of the document where you want to add an annotation.	Query param

Sample response

HTTP Status Code: 201

Response: Annotation is saved successfully!

Getting annotation data

The *Get Annotation Data API* allows you to fetch all the annotations associated with the document.

- **Service URL:** *http or https://<IP address>:<Port>/ecmapi/annotationservice/annotations/saveAnnotation/getAnnotationData*
- **Method type:** POST
- **Request content type:** APPLICATION_FORM_URL_ENCODED_VALUE
- **Response content type:** TEXT_PLAIN_VALUE

Sample Request

While calling the *Get Annotation Data API*, pass the value of the *id* and *documentId* query parameters in the request URL. For example, if you want to fetch the annotations associated with a document having id as 64a3e696e0bc1f048f32900e on the page number with id as 1. Then, pass the request URL as follows:

http or https://<IP address>:<Port>/ecmapi/annotationservice/annotations/getAnnotationData?id=1&documentId=64a3e696e0bc1f048f32900e

The following table lists the descriptions of parameters of the *Get Annotation Data API*:

Parameter	Mandatory/Optional/Conditional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory				String	The document page number where you want to add the annotation.	Form data
documentId	Mandatory		1		String	The unique ID of the document in which you added the annotation.	Form data

Sample response

```
{
  [AnnotationGroupHeader]
  TotalGroups=1
  [Group1]
  Name=Mukesh Dadsena
  LoginUserRights=VM
  AnnotationGroupIndex=0
  [Mukesh DadsenaAnnotationHeader]
  TotalAnnotations=1
  NoOfLine=1
  [Mukesh DadsenaLine1]
  X1=388
  Y1=189
  X2=389
  Y2=189
  Color=255
  TimeOrder=2023,07,17,16,48,24
  MouseSensitivity=1
  AnnotationGroupID=Mukesh Dadsena
  UserID=Mukesh Dadsena
  Rights=VM
  Thickness=1
  Style=0
}
```

Document download service

Document service allows you to download documents from the specific Micro UI repository.

The document service includes the [Downloading documents](#) API.

Headers

The below table lists the header parameters for annotation services:

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
tenantId	Mandatory		String	When a client registers at the NewgenONE Content Cloud platform, a tenant ID is generated. You can copy the tenant ID from the activation email you received while signing in to the NewgenONE Content Cloud platform.
Content-Type	Mandatory	application/json	String	Format of the content in which the request and response parameters need to pass.
accessToken	Mandatory		String	An accessToken is an authentication token obtained from the login API.

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
org	Mandatory	ECM	String	The value of this parameter decides the service to authenticate the request. The default value of this parameter is ECM. Use the same parameter if not instructed otherwise.
userId	Conditional		String	<p>This parameter defines the user ID of the logged-in user. Although userId is required to be received by services in the Web-to-Server scenario. If it is not passed in the request, the application gateway will pick it up from accessToken and add it to the request while routing it to specified requests. As a result, the request does not fail if the userId parameter is not passed.</p> <p>This parameter must be sent since there is no way to access the userId parameter from an accessToken in a Server-to-Server scenario. Otherwise, services may raise an exception.</p>

Parameter	Mandatory/Conditional	Accepted Values	Data Type	Description
appld	Mandatory		String	This parameter defines the App ID of the application through which the accessToken is created using Auth or Token API.
secret	Mandatory		String	This parameter defines the secret code through which the accessToken is created using Auth or Token API.

Downloading documents

The *Download document* API allows you to download documents of any size.

- **Service URL:** *http or https://<IPaddress>:<Port>/ecmapi/documentdownloadservice/document/download/{id}*
- **Method type:** GET
- **Request content type:** application/json
- **Response content type:** application/json

Sample request

While calling the *Download document* API, pass the *id* parameter in the request URL. For example, if you want to download a document whose ID is 62bc4ef9e7c42f246f4ca46a, then pass the request URL as follows:

```
http or https://<IPaddress>:<Port>/ecmapi/documentdownloadservice/document/download/62bc4ef9e7c42f246f4ca46a
```

The following is the description of the request parameters of the *Download document* API:

Parameter	Mandatory/Optional	Accepted Values	Default Value	System Field	Data Type	Description	Parameter Type
id	Mandatory				String	ID of the document.	Path params
contentType	Mandatory				String	Format of the content. For example, PDF, Image, and more.	URL params
docName	Mandatory				String	Name of the document.	Query params
userId	Mandatory				String	This parameter defines the user ID of the logged-in user.	Query params

Sample response

HTTP Status Code: 200

HTTP Status Code Value: OK

Sample code

The following sample code covers a scenario to show the user authentication using code and token APIs, and signing in to the NewgenONE Content Cloud platform to create a folder, upload content, and search content by ID.

```
package com.newgen.samplecode;

import java.io.File;
import java.io.IOException;
import java.security.KeyManagementException;
import java.security.KeyStoreException;
import java.security.NoSuchAlgorithmException;
import java.security.SecureRandom;
import java.security.cert.CertificateException;
import java.security.cert.X509Certificate;
import java.util.List;
import java.util.Map;
import javax.net.ssl.HttpURLConnection;
import javax.net.ssl.SSLContext;
import javax.net.ssl.TrustManager;
import javax.net.ssl.X509TrustManager;
import javax.validation.Validation;
import javax.validation.Validator;
import javax.validation.ValidatorFactory;
import org.apache.http.conn.ssl.NoopHostnameVerifier;
import org.apache.http.conn.ssl.SSLConnectionSocketFactory;
import org.apache.http.conn.ssl.TrustStrategy;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.json.JSONObject;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.annotation.Bean;
import org.springframework.core.io.ClassPathResource;
import org.springframework.core.io.FileSystemResource;
import org.springframework.http.HttpEntity;
import org.springframework.http.HttpHeaders;
import org.springframework.http.HttpMethod;
import org.springframework.http.MediaType;
import org.springframework.http.ResponseEntity;
import org.springframework.http.client.HttpComponentsClientHttpRequestFactory;
import org.springframework.util.LinkedMultiValueMap;
import org.springframework.util.MultiValueMap;
import org.springframework.validation.beanvalidation.MethodValidationPostProcessor;
```

```

import org.springframework.web.client.RestTemplate;
import org.springframework.web.multipart.MultipartFile;
import org.springframework.web.util.UriComponentsBuilder;

@SpringBootApplication
public class SamplecodeApplication {
    public static RestTemplate restTemplate;

    @Bean
    public Validator createValidatorBean() {
        ValidatorFactory factory = Validation.buildDefaultValidatorFactory();
        return factory.getValidator();
    }

    @Bean
    public RestTemplate restTemplate()
        throws KeyStoreException, NoSuchAlgorithmException, KeyManagementException
    {
        TrustStrategy acceptingTrustStrategy = (X509Certificate[] chain, String
authType) -> true;
        SSLContext sslContext = org.apache.http.ssl.SSLContexts.custom()

            .loadTrustMaterial(null, acceptingTrustStrategy)

            .build();
        SSLConnectionSocketFactory csf = new
SSLConnectionSocketFactory(sslContext);
        CloseableHttpClient httpClient =
HttpClientBuilder.create().setSSLSocketFactory(csf).build();
        HttpClientComponents requestFactory = new
HttpClientComponents.Builder().setHttpClient(httpClient).build();
        RestTemplate restTemplate = new RestTemplate(requestFactory);
        return restTemplate;
    }

    @Bean
    public MethodValidationPostProcessor methodValidationPostProcessor() {
        return new MethodValidationPostProcessor();
    }

    static void ignoreCertificates() {
        TrustManager[] trustAllCerts = new TrustManager[] {new X509TrustManager()
    {
        @Override public X509Certificate[] getAcceptedIssuers(){return null;}

        @Override
        public void checkClientTrusted(X509Certificate[] certs, String
authType) {}
        @Override
        public void checkServerTrusted(X509Certificate[] certs, String

```

```

authType) {}
    });
    try {
        SSLContext sc = SSLContext.getInstance("TLS");
        sc.init(null, trustAllCerts, new SecureRandom());
        HttpURLConnection.setDefaultSSLSocketFactory(sc.getSocketFactory());

    } catch (Exception e) {}
}

public static void main(String[] args) throws IOException {
    SpringApplication.run(SamplecodeApplication.class, args);
    // getting code for authentication
    String url = "https://devcontentservices.newgensoftware.net:8197/ecmapi/
appservice/auth/code";
    UriComponentsBuilder builder = UriComponentsBuilder.fromUriString(url);

    HttpHeaders headers = new HttpHeaders();
    headers.setContentType(MediaType.APPLICATION_JSON);
    // headers.add("securityClearance", securityClearance);
    headers.add("userId", "");
    headers.set("appId", "");
    headers.set("tenantId", "");
    headers.set("authRequired", "false");
    headers.set("org", "ECM");
    String body = "{\n"
        + "\"emailId\": \"user@mailinator.com\", \n"
        + "\"redirect_uri\": \"https://ecmnext.azurewebsites.net/\", \n"

        + "\"password\": \"\" \n"
        + "}";
    HttpEntity<String> requests = new HttpEntity<>(body, headers);
    ignoreCertificates();
    RestTemplate restTemplate = new RestTemplate();
    ResponseEntity<String> response = restTemplate.exchange(
        builder.toUriString(), HttpMethod.POST, requests, String.class);
    JSONObject responses = new JSONObject(response.getBody());
    System.out.println(responses);
    int codes = responses.get("redirect_uri").toString().indexOf("?") + 6;
    String code = responses.get("redirect_uri").toString().substring(codes);

    System.out.println(code);
    headers.set("secret", "");
    // calling token api by using above code api to get required authorization

    // token which can be used in other api like accessToken.
    String urls = "https://devcontentservices.newgensoftware.net:8197/ecmapi/
appservice/auth/token";
    UriComponentsBuilder buildera = UriComponentsBuilder.fromUriString(urls);

    JSONObject d = new JSONObject();

```

```

d.put("code", code);
headers.set("type", "CODE");
HttpEntity<String> requestq = new HttpEntity<>(d.toString(), headers);
ResponseEntity<String> responsesd = restTemplate.exchange(
    buildera.toUriString(), HttpMethod.POST, requestq, String.class);
JSONObject responsesdd = new JSONObject(responsesd.getBody());
headers.set("accessToken", responsesdd.getString("accessToken"));
JSONObject dl = new JSONObject();
dl.put("emailId", "user@mailinator.com");
dl.put("password", "<enter the password>");
// after getting accessToken from above api,now trying to login by using
below
// api
String urla = "https://devcontentservices.newgensoftware.net:8197/ecmapi/
userregisterservice/users/login";
HttpEntity<String> requestqd = new HttpEntity<>(dl.toString(), headers);

UriComponentsBuilder builderad = UriComponentsBuilder.fromUriString(urla);

ResponseEntity<String> responsesds = restTemplate.exchange(
    builderad.toUriString(), HttpMethod.POST, requestqd, String.class);

String bodys = "{\n"
    + "    \"folderName\": \"\", \n"
    + "    \"folderType\": \"folder\", \n"
    + "    \"comments\": \"My workspace for project A\", \n"
    + "    \"ownerName\": \"\", \n"
    + "    \"ownerId\": \"\", \n"
    + "    \"parentFolderId\": \"\", \n"
    + "    \"dataclass\": {\n"
    + "        \"dataClassName\": \"\", \n"
    + "        \"dataClassId\": \"\", \n"
    + "        \"clientNames\": \"sq\" \n"
    + "    }, \n"
    + "    \"metadata\": {\"key\": \"v\"}, \n"
    + "    \"securityClass\": [\"OWNER\"] \n"
    + "}";
// after successful login trying to create a folder in a root folder
String urlad = "https://devcontentservices.newgensoftware.net:8197/ecmapi/
folderservice/folders";
HttpEntity<String> requestqds = new HttpEntity<>(bodys, headers);
UriComponentsBuilder builderads =
UriComponentsBuilder.fromUriString(urlad);
ResponseEntity<String> responsesdss = restTemplate.exchange(
    builderads.toUriString(), HttpMethod.POST, requestqds, String.class);

System.out.println(responsesdss);
// after creating folder,below api is adding content/document in the above

// created folder
String urlx = "https://devcontentservices.newgensoftware.net:8197/ecmapi/

```

```

contentservice/contents/sync";
    HttpHeaders header = new HttpHeaders();
    header.setContentType(MediaType.MULTIPART_FORM_DATA);
    header.add("userId", "");
    header.set("appId", "");
    header.set("tenantId", "");
    header.set("authRequired", "false");
    header.set("org", "ECM");
    header.set("accessToken", responsesdd.getString("accessToken"));
    header.set("secret", "");
    LinkedMultiValueMap<Object, Object> map = new LinkedMultiValueMap<>();
    JSONObject f = new JSONObject();
    map.add("name", "<enter the content name>");
    map.add("contentType", "pdf");
    map.add("parentFolderId", "");
    map.add("storageCredentialId", "");
    map.add("ownerName", "<enter the owner name>");
    map.add("ownerId", "<enter the owner ID>");
    map.add("documentType", "pdf");

```

Common error codes

The following table describes the common error codes and their description related to NewgenONE Content Cloud Microservices:

Error code	Error message
10001	Invalid Tenant ID.
10002	Content not found.
10003	Constraint violation.
10004	Content location not found.
10005	Storage location not found.
10006	Folder not found.
10007	Lock already exists.
10008	Unknown error.
10009	Invalid object type.
100010	Invalid privilege type.

Error code	Error message
100011	Version conflict.
100012	Parent folder not found.
100013	Target folder not found.
100014	Failed to create content location.
100015	Parent folder marked for deletion.
100016	Database connection failure.
100017	Storage credential not found.
100018	Storage process not found.
100019	Folders and content associated.
100020	No results found.
100021	Email ID is not Valid.
100022	Password is missing.
100023	Parameter is missing.
100025	Unauthorized User.
100026	Invalid User ID.
100027	Cabinet not found.
100028	Invalid Token.
100029	System-generated folders and files cannot be updated.
100030	Datatype mismatch.
100031	Database connection failure.
100032	Lock already exists.
100033	Requested HTTP method not supported for this API.
100034	Search params null or invalid.
100035	Error generating key.