

## Cloudinary Magento 2 Extension User Guide

Add the Cloudinary Magento 2 Extension to manage your images and videos through Cloudinary. You can apply transformations to change the visual appearance or resize to support multiple devices, and deliver your assets in the optimal format and quality via the most advanced Content Delivery Networks (CDNs). The Cloudinary Magento extension now includes full support for Magento Page Builder functionality, making it easy to build your site content and use your Cloudinary media.

### Make sure you have a Cloudinary account

A Cloudinary account is required to use this extension. [Sign up](#) for a free plan, starting with generous usage limits and when your requirements grow, you can easily upgrade to a plan that best fits your needs. More pricing information available [here](#).

### Configure the Extension

Once the extension is downloaded and installed, follow these steps:

1. In the **Magento Admin Panel**, select 'Stores > Configuration'.
2. On the **Configuration** page, open the 'Cloudinary' section in the left-hand sidebar and select 'Settings'.
3. In the **Cloudinary Setup** section of this page:
  - a. Set your 'Cloudinary Account Credentials' (copy the **Environment variable** as found in the [Cloudinary Management Console](#)). The format should be `cloudinary://{API_Key}:{API_Secret}@{Cloud_Name}`.
  - b. Set 'Enable Cloudinary' to **Yes**.
  - c. Further options can be specified in the 'Cloudinary Configuration' and 'Default Image Transformations' sections of the Cloudinary config (see the [Cloudinary Features](#) section below).
  - d. Click the **Save Config** button at the top of the page.
4. Go to 'System > Cache Management' and refresh your **Configuration** and **Page Caches**.

From this point onwards, any new media assets you upload will be delivered via Cloudinary.

### Migrate Media to Cloudinary

To serve your existing assets via Cloudinary you will need to migrate them first. You can do this in two different ways:

- Enable **Auto Upload Mapping** to automatically upload existing media assets to Cloudinary when they are first requested by a user. This option can be set in the **Cloudinary Configuration** section of the Cloudinary settings.

or

- Migrate your media in bulk using the **Magento command line tool**:
  1. From the command line, run the `cloudinary:upload:all` command. If you are currently located within Magento's parent directory, the correct command is: `bin/magento cloudinary:upload:all`.
  2. You can follow the progress of the migration from the command line. Keep your terminal window open while the uploads are in progress.
  3. When all media assets have been uploaded, return to 'System > Cache Management' in the **Magento Admin Panel** and refresh all caches.

## Cloudinary Features

With Cloudinary, you can add and deliver both images and videos. The Cloudinary configuration page has multiple options to change the behaviour of the Cloudinary extension when serving images and video. These are:

- **Media Delivery Domain Sharding** - Enables multiple subdomains for Cloudinary delivery URLs in order to increase page loading speed. This is enabled by default.
- **Auto Upload Mapping** - Automatically upload media assets from your site to Cloudinary (if they don't already exist in your account). Use this to lazily migrate your assets, as the upload will occur only when the media is first requested by a user.
- **Automatic Image Format Optimization** - Automatically deliver images converted to modern image formats based on viewing device and browser. For example, deliver WebP on Chrome and JPEG-XR on Internet Explorer for better performance and user experience. This is enabled by default.
- **Image Quality** - Adjust quality of generated images to balance between visual quality and file size minimization. The quality setting defaults to use Cloudinary's automatic quality algorithm. This is particularly relevant for JPEG and WebP compression levels.
- **Image Cropping Gravity** - Define the part of the image to focus on when cropping images in order to better match your graphic design. This setting will have no effect unless an option other than 'Magento's Default' is chosen.
- **Image Device Pixel Ratio (DPR)** - Use a DPR value higher than 1.0 to generate and deliver hi-res images for better visual results on HiDPI devices, such as Retina Display devices. This is set to 1.0 by default, but can be increased to 2.0.
- **Global custom transformations** - Apply additional transformations to every image and video thumbnail you deliver. For example, you could add a semi-transparent company logo watermark to every asset. Specify the transformation parameters in the URL format as described in Cloudinary's [image transformations guide](#).
- **Lazy Loading** - Only load your images when scrolled into view.

- Configure whether to apply to your CMS blocks, and optionally, which blocks to exclude from lazy loading.
- Set the threshold for how many pixels from the top of the page before lazy loading is applied to your images.
- Choose an effect for how the images are loaded.
- Choose a placeholder type such as blur or pixelate.
- **Product Gallery** - Add Cloudinary's interactive media gallery interface to deliver your optimized product images and videos. See the Product Gallery [guide](#) for more information.
  - Select the color scheme for the gallery to match your brand.
  - Configure the navigation, zoom type and position, and the behavior of the transitions between images.
  - Configure all aspects of the carousel, including whether to use thumbnails or indicators as well as the thumbnail size and presentation.
  - Extend the functionality by configuring custom parameters. See [product gallery reference](#).
  - Add [360 spin sets](#) to your gallery when selecting product images. In order to display a 360 spin set, the Product Gallery fetches all images with a particular tag and combines them into a single entity, where each image is included as a single frame of the resulting 360 spin set (sorted alphanumerically by their Public ID).


When adding a spin set, specify the tag name used for the images, as shown below:

Add Spinset from Cloudinary ✕

Cancel Save

Cloudinary Spinset Tag

Preview



## Cloudinary Media Library

Enjoy a full Digital Asset Management (DAM) solution directly through Magento. Use the Cloudinary Media Library UI to:

- Upload new images, videos, audio, or other files
- Create and navigate through folders or share folders with user groups
- Create and browse collections or share collections with user groups or external parties
- Perform advanced searches
- Select multiple assets to perform bulk operations including delete, add/edit tags, move, download, or review moderated assets.
- Select a single asset to view or update basic asset details, view or activate advanced asset analysis operations, or to generate transformations for that asset.

See the [Cloudinary DAM documentation](#) for more information.

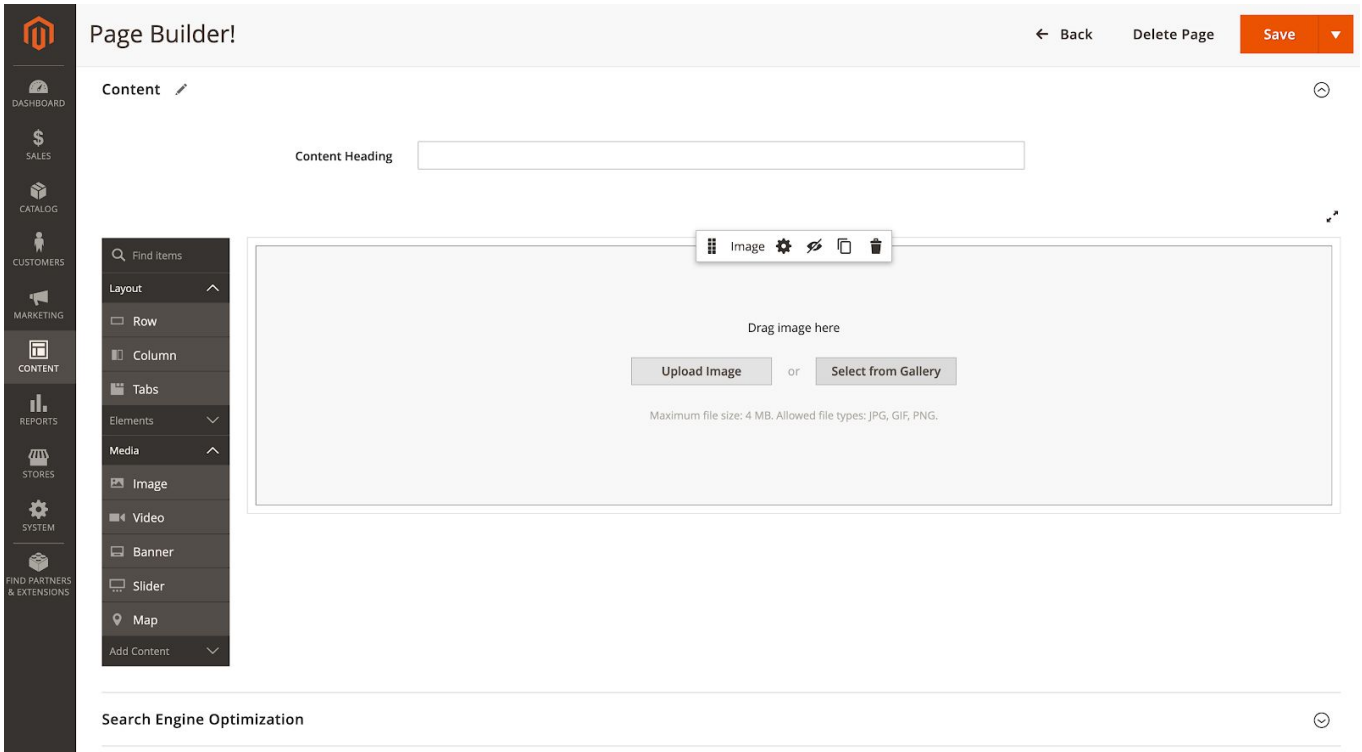
Look out for the “Add from Cloudinary” button to launch the Media Library. You can access it in the following places:

- **Managing your catalog** - Add category and product images to your catalog directly from the Cloudinary Media Library.
- **Managing your site content** - Add media from Cloudinary to all pages on your Magento site using the Media Library.

## Magento Page Builder

If you're using the Magento Page Builder to manage your site content, you can easily insert and manage your media using Cloudinary. Whenever you add a media block to your site, such as an image, video or banner, you'll have the option to select the media from Cloudinary.

For example, to add an image from Cloudinary, drag and drop an image or banner block into the page builder then click **Select from Gallery**. From there, open up the Cloudinary Media Library to manage and insert your Cloudinary assets by clicking **Add from Cloudinary**. For video blocks, add the Cloudinary URL for the video, including any [transformations](#).



## Transformations

The Clouinary Media Library allows you to perform transformations on your individual assets before inserting them to your Magento site. Have a look at the [image transformations guide](#) for more information on the available transformations.

**Note:** If you have global transformations defined, these will be combined with the transformations you perform at an individual level.

## Product Catalog API

You can use the Clouinary product catalog API to programmatically link your Clouinary media to your products by providing the URL and product SKU.

## Configuration

To configure the API, ensure you have the Clouinary plugin installed, as described above. The API is built on top of the Magento 2 REST API and therefore you'll need to create a new integration in order to allow access to edit the product catalog via the API.

To create the integration:

1. In the Magento Admin Panel, select **System > Integrations**.
2. Select **Add New Integration**.
3. In the **Integration Info** tab, enter the relevant information as well as your username and password.
4. In the **API** tab, under **Catalog > Inventory > Products** select **Edit Product Design** to give the integration access to edit the product media.

## Available APIs

Resource Access

Resources

- Dashboard
- Analytics
  - API
- Sales
- Catalog
  - Inventory
    - Products
      - Edit Product Design
    - Categories
      - Edit Category Design
- Customers
- Carts
- My Account
- Marketing
- Content
- Reports
- Stores
- System
- Global Search

- Once added, select **Activate** and allow access to the selected APIs.
- Retrieve your **Access Token**, you will use this to authenticate your requests to the API.

## Authentication

The API uses **Bearer Authentication**. Send the token you retrieved when creating your new integration. An example header would be:

```
Authorization: Bearer abc123mytoken123
```

The `Content-Type` header is also required for all requests. This will always be `application/json`. For example:

```
Content-Type: application/json
```

## Syntax

As the API is built on top of the Magento REST API, the base URL is the same:

```
https://{{magentoUrl}}/rest/V1
```

The Clouinary plugin creates one new endpoint:

```
https://{{magentoUrl}}/rest/V1/cloudinary/products/{SKU}/media
```

Include the product `SKU` in the URL, this is the product you want to add your media to.

Send the Clouinary URLs for the media you want to add in the JSON request body.

## Parameters

Parameter	Type	Description
<code>urls</code>	Array of URL objects	An array of objects, each containing Clouinary URLs (and optionally public IDs) to add to a product. <b>Required.</b>

## URL object parameters

Parameter	Type	Description
<code>publicId</code>	String	The Clouinary Public ID of the product media. Only required if the Clouinary URL does not contain a <a href="#">version</a> . <b>Required.</b>
<code>url</code>	String	The Clouinary URL of the product media. <b>Optional.</b>
<code>roles</code>	String or Array	The product image role(s) of the image to be added to a product. Can be either an array of strings or a single string containing a comma separated list. <b>Optional.</b>
<code>label</code>	String	The label associated with the image. Used as the alt text. <b>Optional.</b>
<code>disabled</code>	Boolean	Whether the image should be hidden from the product page. <b>Default:</b> false. <b>Optional.</b>

## Examples

Example of adding a sample image and video to the product SKU “amazing-product-sku”:

<https://amazing-magento-shop.com/rest/V1/cloudinary/products/amazing-product-sku/media>

```
{
  "urls": [
    {
      "url":
"https://res.cloudinary.com/demo/image/upload/v123/sample.jpg"
    },
    {
      "url":
"https://res.cloudinary.com/demo/video/upload/v123/dog.mp4"
    }
  ]
}
```

Example of adding a sample image and video with no version provided:

```
{
  "urls": [
    {
      "url":
"https://res.cloudinary.com/demo/image/upload/sample.jpg",
      "publicId": "sample"
    },
    {
      "url":
"https://res.cloudinary.com/demo/video/upload/dog.mp4",
      "publicId": "dog"
    }
  ]
}
```

Example of adding a sample image to the product SKU “amazing-product-sku” with a label, and the roles: image, small\_image and thumbnail:



```
{
  "urls": [
    {
      "url":
"https://res.cloudinary.com/demo/image/upload/v123/sample.jpg",
      "roles": ["image","small_image","thumbnail"],
      "label": "Amazing Product",
      "disabled": false
    }
  ]
}
```

### Sample Response

```
"{\\"passed\\":2,\\"failed\\":{\\"count\\":0,\\"urls\\":[]}}"
```

FAQs can be found in Clouddinary's [knowledge base](#).

If you have any questions about this extension or the Clouddinary service, feel free to [contact](#) Clouddinary's support team.