



**PLATFORM DEEP DIVE / Core components / Core extensions / Content  
management / media-library**

# Contents

- PLATFORM DEEP DIVE / Core components / Core extensions / Content management / Media library
  - Displaying assets
  - Searching assets
  - Replacing assets
  - Referencing assets in UI Designer
  - Icons
    - Customization
  - Export/import media assets
    - Import media assets
    - Export all

## PLATFORM DEEP DIVE / Core components / Core extensions / Content management / Media library

The media library serves as a centralized hub for managing and organizing various types of media files, including images, GIFs, and more. It encompasses all the files that have been uploaded to the

The fallback content to display on prerendering  
, providing a convenient location to view, organize, and upload new media files.

Preview	Key	Format	Size	Edited at	Edited by
<input type="checkbox"/>	silviu	webp	0.51 MB	04 Oct 2022, 11:29 AM	John Doe
<input type="checkbox"/>	teamwork	png	0.1 MB	04 Oct 2022, 9:24 AM	John Doe
<input type="checkbox"/>	dodge-challenger	jpeg	0.19 MB	04 Oct 2022, 9:24 AM	John Doe
<input type="checkbox"/>	viper_2	jpeg	0.82 MB	04 Oct 2022, 9:24 AM	John Doe
<input type="checkbox"/>	dodge viper	jpeg	0.05 MB	04 Oct 2022, 9:24 AM	John Doe
<input type="checkbox"/>	impala	gif	0.74 MB	04 Oct 2022, 9:24 AM	Jane Doe
<input type="checkbox"/>	lamborghini	png	0.68 MB	04 Oct 2022, 9:24 AM	John Doe

### ! INFO

You can also upload an image directly to the Media Library on the spot when configuring a process using the **UI Designer**. More information [here](#).

## Uploading a new asset

To upload an asset to the Media Library, follow the next steps:

### 1. Open

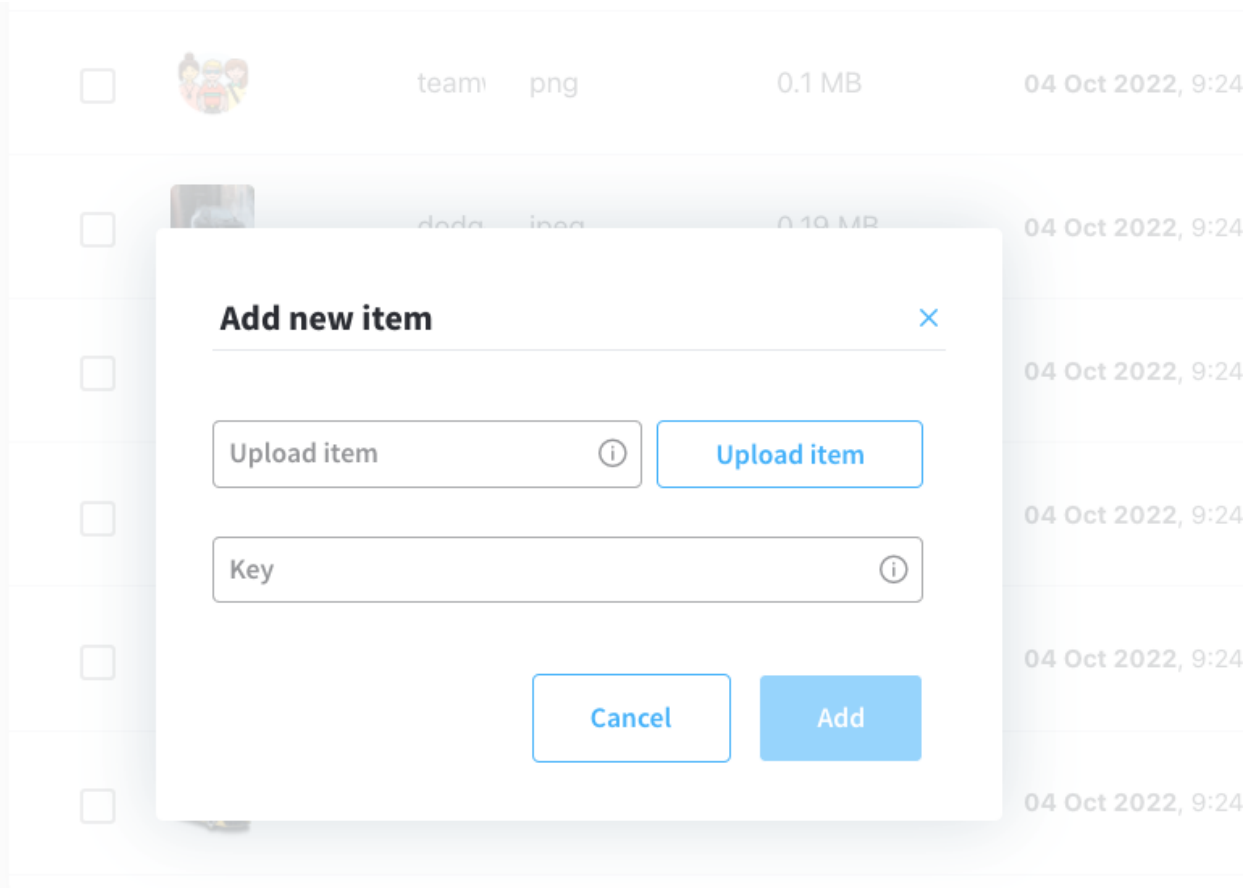
The fallback content to display on prerendering

.

### 2. Go to **Content Management** tab and select **Media Library**.

3. Click **Add new item**, the following details will be displayed:

- **Upload item** - opens a local file browser
- **Key** - the key must be unique, you cannot change it afterwards



4. Click **Upload item** button and select a file from your local browser.

5. Click **Upload item** button again to upload the asset.

#### CAUTION






















Supported formats: PNG, JPEG, JPG, GIF, SVG or WebP format, 1 MB maximum size.

## Displaying assets

Users can preview all the uploaded assets just by accessing the **Media Library**.

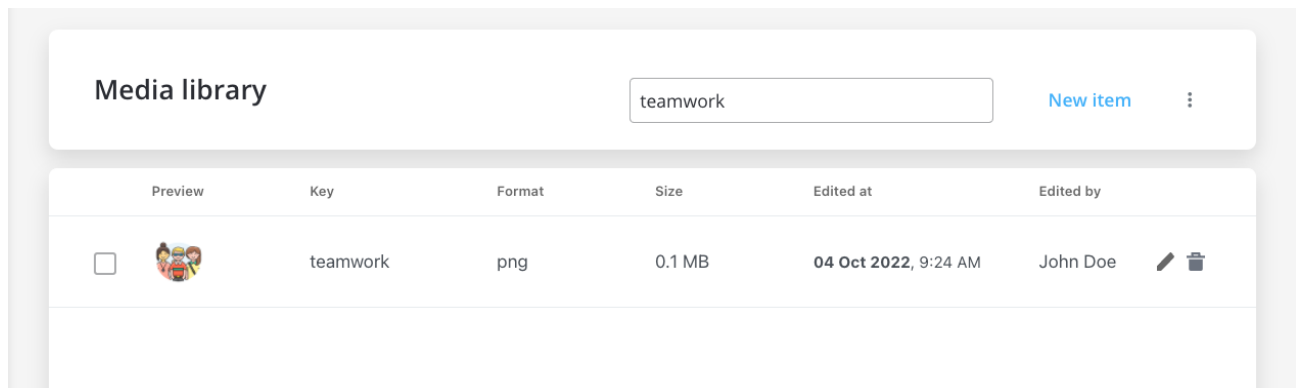
You have the following information about assets:

- Preview (thumbnail 48x48)
- Key
- Format ("- " for unknown format)
- Size
- Edited at
- Edited by

Media library							
		Search item by key				New item	
Preview	Key	Format	Size	Edited at	Edited by		
<input type="checkbox"/> 	silviu	webp	0.51 MB	04 Oct 2022, 11:29 AM	John Doe		
<input type="checkbox"/> 	teamwork	png	0.1 MB	04 Oct 2022, 9:24 AM	John Doe		
<input type="checkbox"/> 	dodge-challenger	jpeg	0.19 MB	04 Oct 2022, 9:24 AM	John Doe		
<input type="checkbox"/> 	viper_2	jpeg	0.82 MB	04 Oct 2022, 9:24 AM	John Doe		
<input type="checkbox"/> 	dodge viper	jpeg	0.05 MB	04 Oct 2022, 9:24 AM	John Doe		
<input type="checkbox"/> 	impala	gif	0.74 MB	04 Oct 2022, 9:24 AM	John Doe		
<input type="checkbox"/> 	lamborghini	png	0.68 MB	04 Oct 2022, 9:24 AM	John Doe		

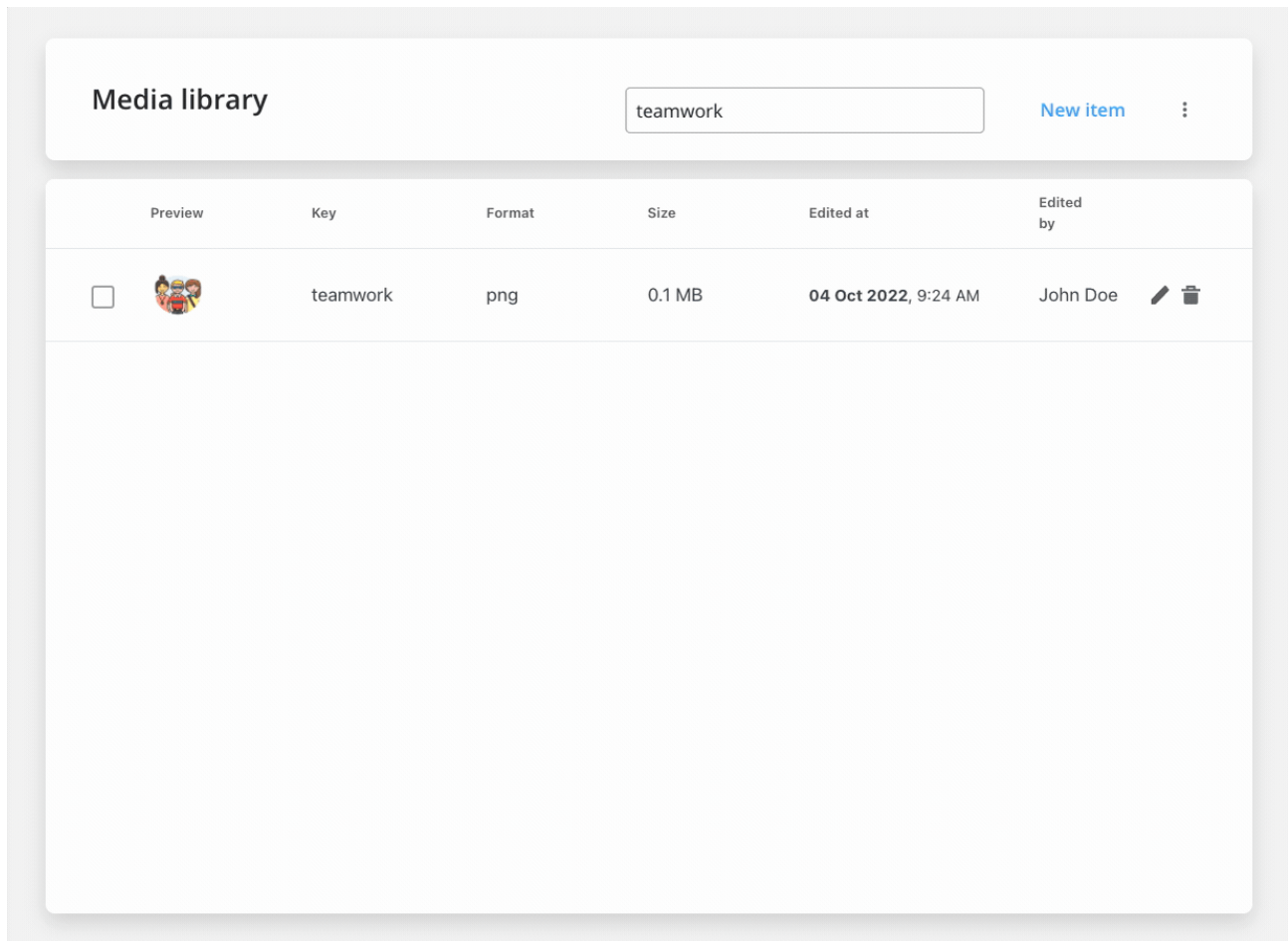
## Searching assets

You can search an asset by using its key (full or substring).



## Replacing assets

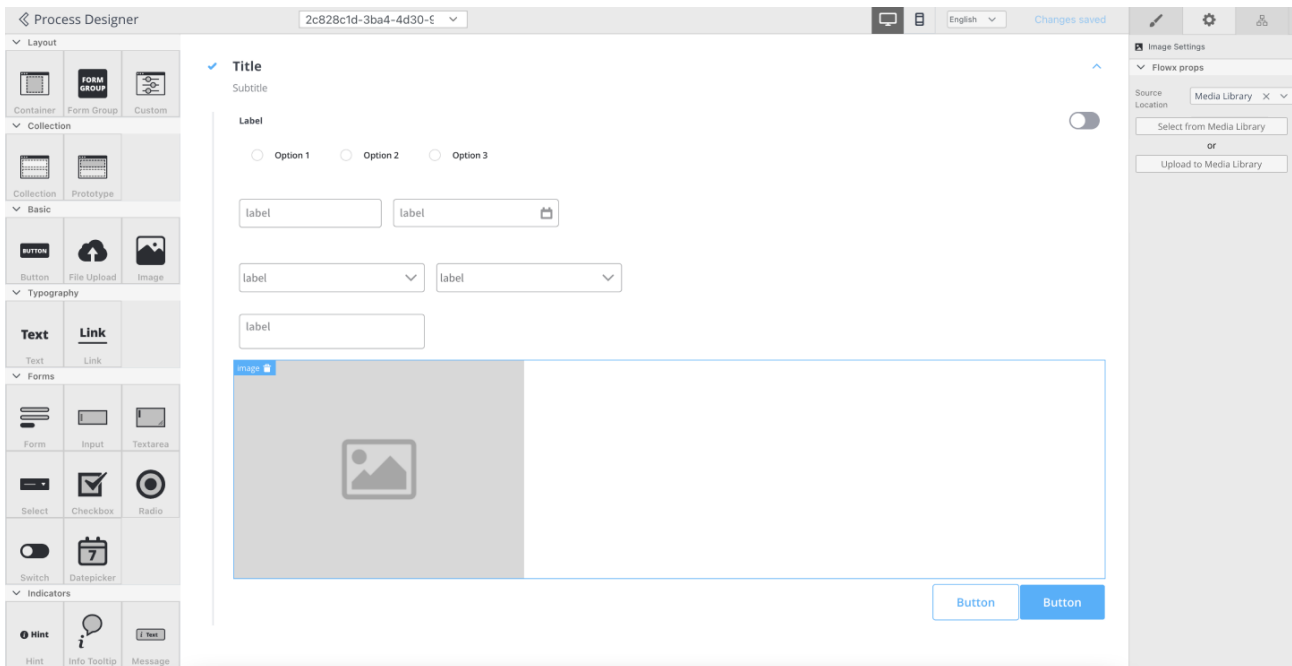
You can replace an item on a specific key (this will not break references to process definitions).



## Referencing assets in UI Designer

You have the following options when configuring image components using **UI Designer**:

- Source Location - here you must select **Media Library** as source location
- Image Key
  - **Option 1**: trigger a dropdown with images keys - you can type and filter options or can select from the initial list in dropdown
  - **Option 2**: open a popup with images thumbnails and keys then you can type and filter options or can select from the initial list



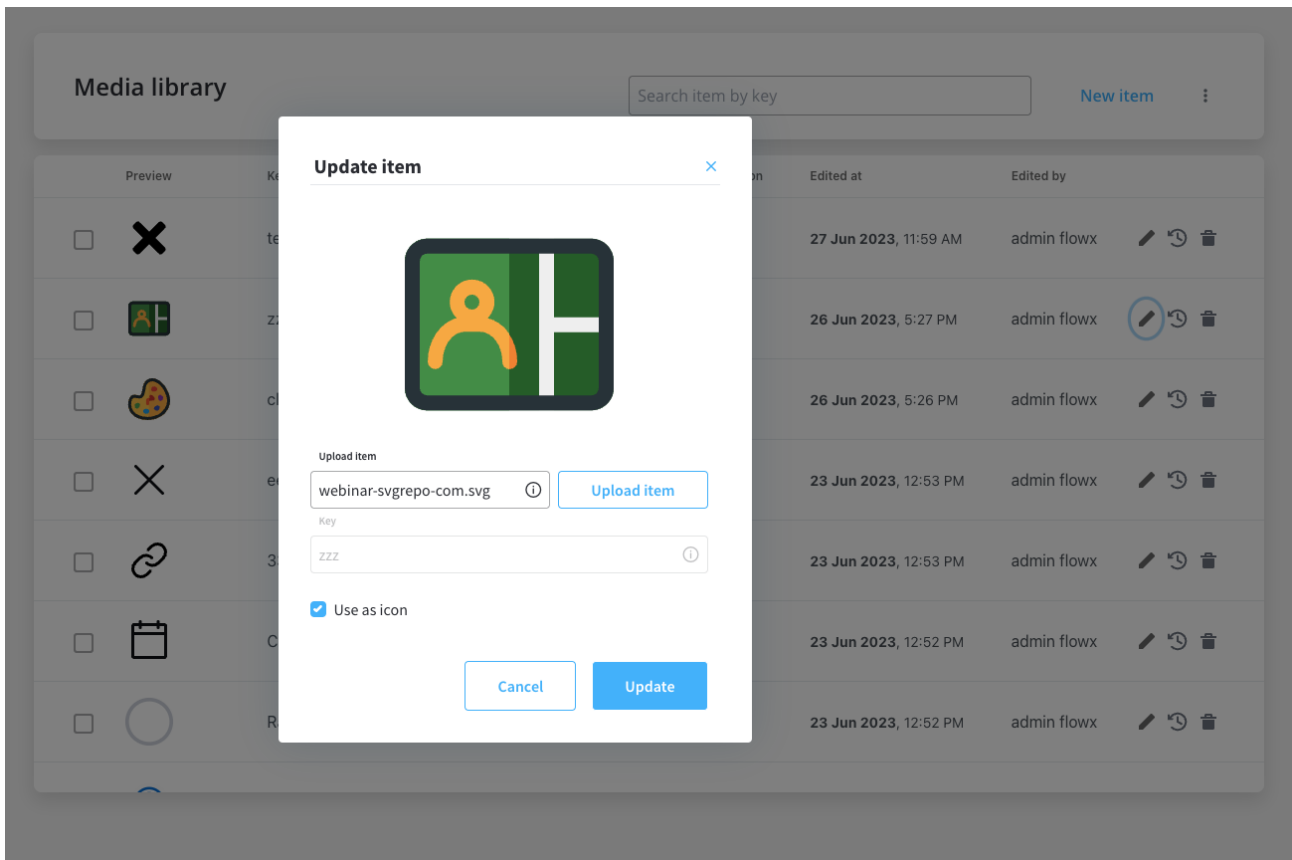
### ! INFO

More details on how to configure an image component using UI Designer - [here](#).

## Icons

The Icons feature allows you to personalize the icons used in UI elements. By uploading SVG files through the Media Library and marking them, you can choose icons from the available list in the UI Designer.





### ! INFO

When selecting icons in the UI Designer, only SVG files marked as icons in the Media Library will be displayed.

### ! INFO

To ensure optimal visual rendering and alignment within your UI elements, it is recommended to use icons with small sizes such as: 16px, 24px, 32px.

Using icons specifically designed for these sizes helps maintain consistency and ensures a visually pleasing user interface. It is advisable to select icons

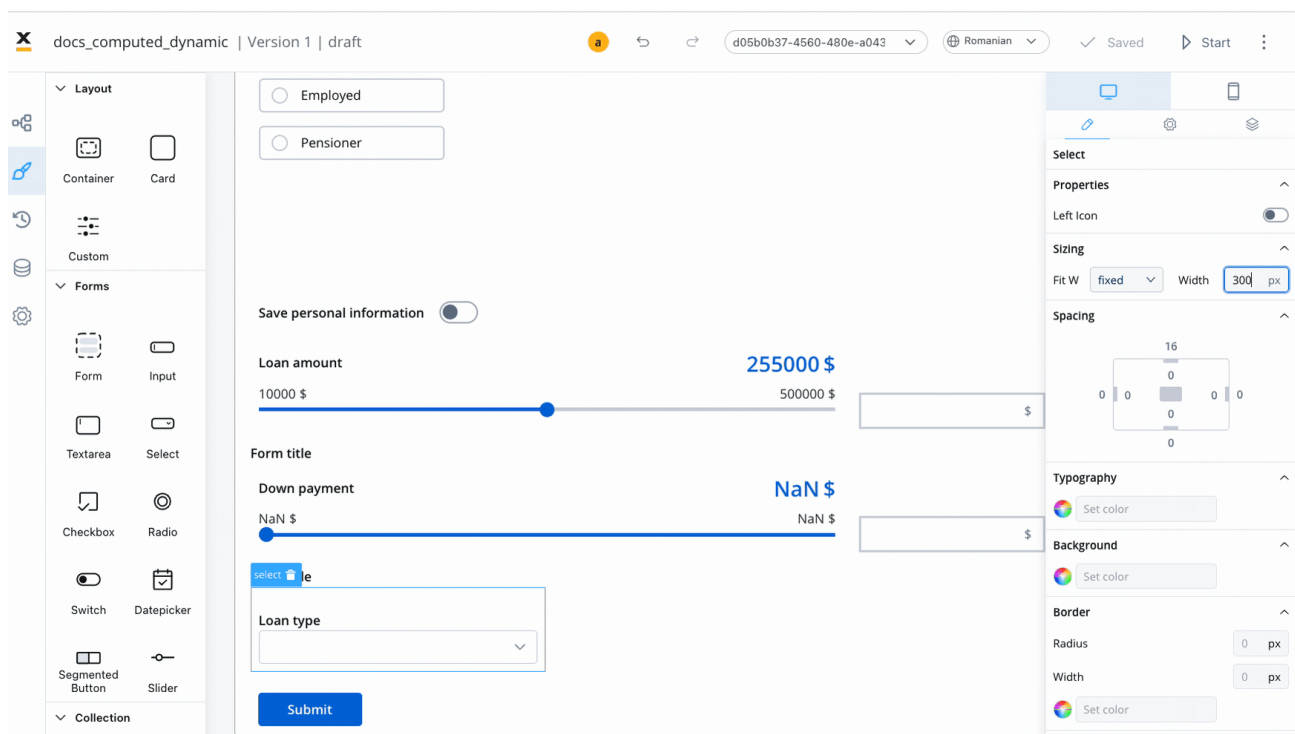
from icon sets that provide these size options or to resize icons proportionally to fit within these dimensions.

### ⚠ CAUTION

Icons are displayed or rendered at their original, inherent size.

## Customization

Content-specific icons pertain to the content of UI elements, such as icons for **input fields** or **send message buttons**. These icons are readily accessible in the **UI Designer**.

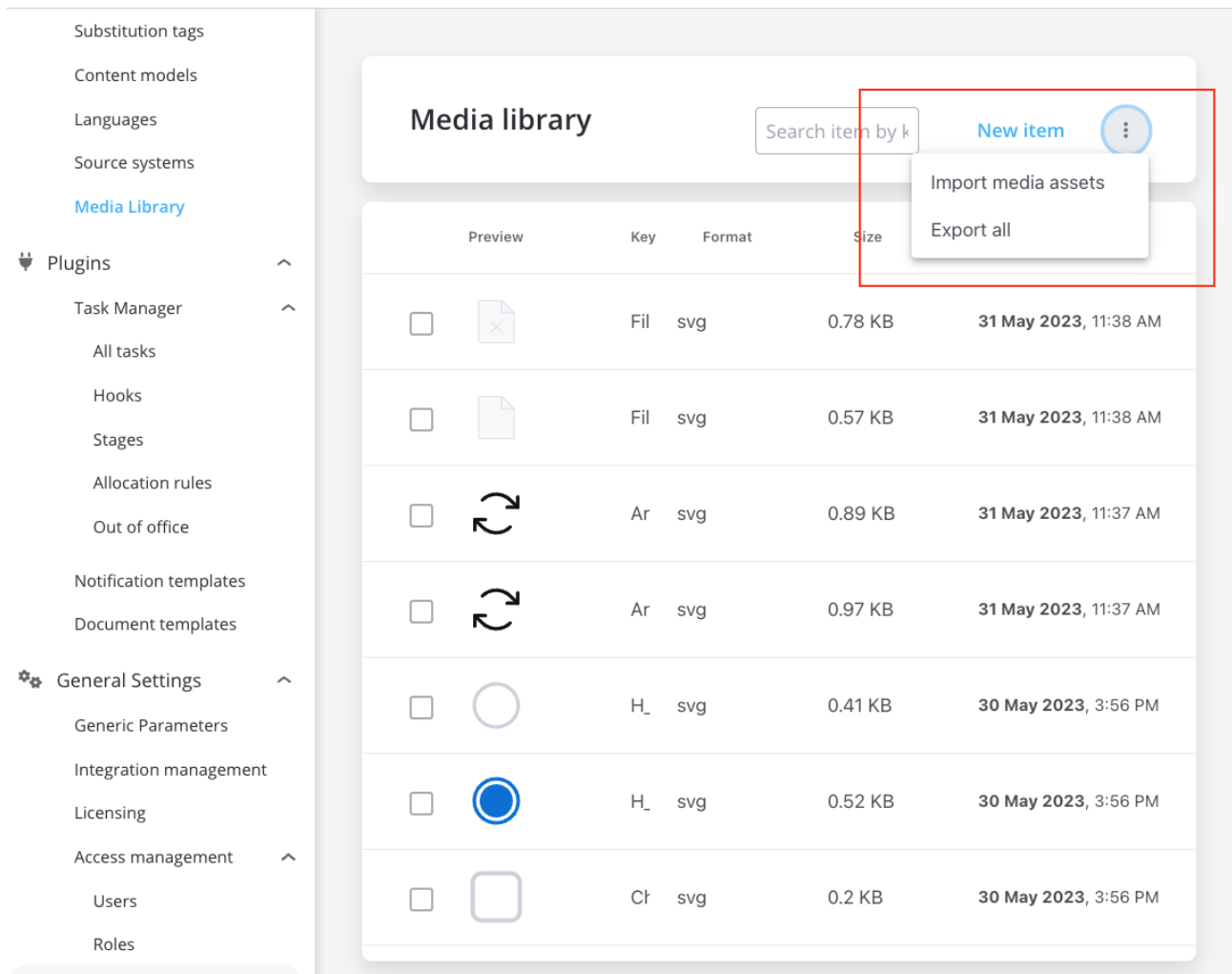


More details on how to add icons on each element, check the sections below:

[» Input element](#)[» Select element](#)[» Buttons](#)

## Export/import media assets

The import/export feature allows you to import or export media assets, enabling easy transfer and management of supported types of media files.



## Import media assets

Use this function to import media assets of various supported types. It provides a convenient way to bring in images, videos, or other media resources.

## Export all

Use this function to export all media assets stored in your application or system. The exported data will be in JSON format, allowing for easy sharing, backup, or migration of the media assets.

The exported JSON structure will resemble the following example:

```
{
  "images": [
    {
      "key": "cart",
      "application": "flowx",
      "filename": "maxresdefault.jpg",
      "format": "jpeg",
      "contentType": "image/jpeg",
      "size": 39593,
      "storagePath":
        "https://d22tnndi9lo60.cloudfront.net/devmain/flowx/cart/16819",
      "thumbnailStoragePath":
        "https://d22tnndi9lo60.cloudfront.net/devmain/flowx/cart/16819",
    },
    {
      "key": "pizza",
      "application": "flowx",
      "filename": "pizza.jpeg",
      "format": "jpeg",
      "contentType": "image/jpeg",
      "size": 22845,
      "storagePath":
        "https://d22tnndi9lo60.cloudfront.net/devmain/flowx/pizza/16819",
      "thumbnailStoragePath":
        "https://d22tnndi9lo60.cloudfront.net/devmain/flowx/pizza/16819",
    }
  ],
  "exportVersion": 1
}
```

- `images` - is an array that contains multiple objects, each representing an image
- `exportVersion` - represents the version number of the exported data, it holds the image-related information
- `key` - represents a unique identifier or name for the image, it helps identify and differentiate images within the context of the application
- `application` - specifies the name or identifier of the application associated with the image, it indicates which application or system the image is related to
- `filename` - the name of the file for the image, it represents the original filename of the image file
- `format` - a string property that specifies the format or file extension of the image
- `contentType` - the MIME type or content type of the image, it specifies the type of data contained within the image file
- `size` - represents the size of the image file in bytes, it indicates the file's storage size on a disk or in a data storage system
- `storagePath` - the URL or path to the location where the original image file is stored, it points to the location from where the image can be accessed or retrieved
- `thumbnailStoragePath` - the URL or path to the location where a thumbnail version of the image is stored, it points to the location from where the thumbnail image can be accessed or retrieved

**Was this page helpful?**