Media field with library

- This section shows how to add a media field to your entity, using the media library, and to only show media items you have uploaded.
- Enable the core **media** module as well as the core **media library** module:

```
bash$ drush en media media_library
The following module(s) will be enabled: media, media_libraty,
image

Do you want to continue? (yes/no) [yes]:
> yes

[success] Successfully enabled: media, media_library, image
```

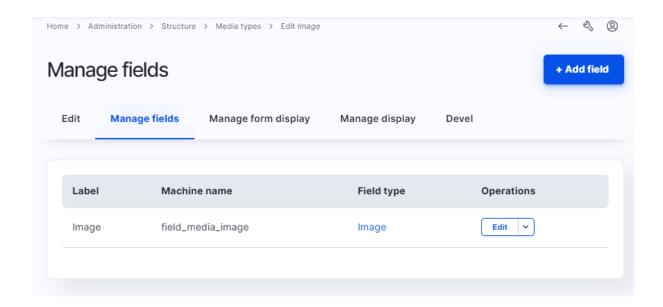
A real advantage of drupal is that time-consuming development practices like file uploads and managing a file library are easily installed with some clicks.

Define a media type at <u>Structure > Media types > Add media Type</u> and choose this:

Name: Image

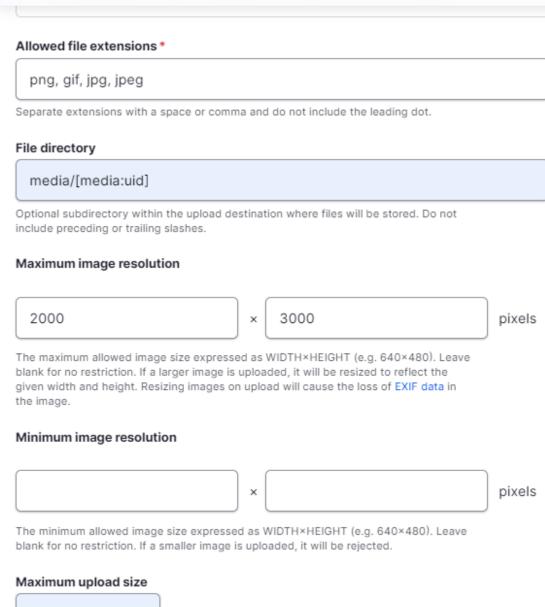
Media source: Image

After saving the media type, you'll discover 'Image' is a bundle of entity media. It has fields which are preconfigured at <u>admin/structure/media/manage/image/fields</u>. In this case, the entity has an image field that is shipped with the media source: image.



We tweak the image field at /damin/structure/media/manage/image/fields/media.image.field_media_image with some settings, so resizing is automatically done when a file is too large.

Image settings for Image



5 MB

Enter a value like "512" (bytes), "80 KB" (kilobytes) or "50 MB" (megabytes) in order to restrict the allowed file size. If left empty the file sizes will be limited only by PHP's maximum post and file upload sizes (current limit 64 MB).



Enable Alt field

Short description of the image used by screen readers and displayed when the image is not loaded. Enabling this field is recommended.

We'll add a media field for our offer at <u>/admin/structure/offer/fields</u> > add field

Add a new field (Media)

Add a label (Image)

Help text (Market your offer with a tremendous image!)

Reference type (Default)

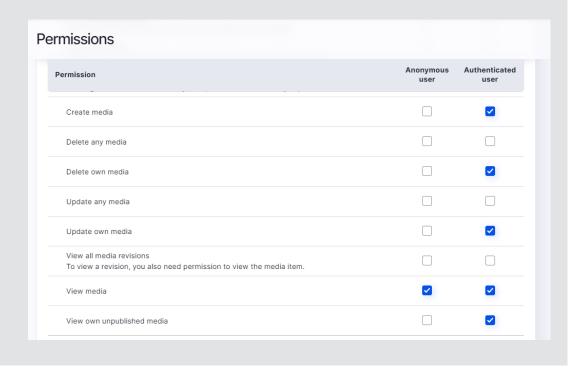
Media Type (Image)

(save and continue)

Allowed number of values (1)

Required field (yes)

An additional step is giving our authenticated users the permissions to add these media items. Because the core media module works like our offer entity, it has its own permissions. Set them to make sure users can create and edit media entities:



Save and head back to our offer create form (<u>/offers/create</u>) we now have a widget with a media library that works for us out-of-the box!

Image No media items are selected. Add media Market your offer with a tremendous image! One media item remaining.

You'll notice that when adding media items to your library, also media items of others are visible. This is because the media library is using a view to display the content. At /add the same contextual filter as we did with the offers view: contextual filters > media:authored by > provide default value > User ID from logged in user.

Important! Do this for every display mode:

/admin/structure/views/view/media_library/edit/page
/admin/structure/views/view/media_library/edit/widget
/admin/structure/views/view/media_library/edit/widget_table

This will only make sure we don't see media items from other users inside our library. Media entities are not secured further in this course. If you would like more security, you could overwrite the Media module AccesControlHandler() by adding a offer_entity_access() hook to your modules/custom/offer/offer.module. You could also use private files for uploads.

But we'll leave this public, as this is not a feature that is needed.