

## **File Managers (python 3):**

### **Django Filer:**

- Show files.
- Upload/copy/cut/rename/delete/show/open files.
- Drag and drop
- Cannot get files from various upload sources

### **Django-filebrowser:**

- browse directories on your server and upload/delete/edit/rename files.
- include images/documents to your models/database using the FileBrowseField.
- select images/documents with TinyMCE.

### **Django-s3direct:**

- Upload/remove/edit files.
- Cannot copy/drag and drop files.

### **Django-attachments :**

- adding attachments
- deleting attachments
- deleting foreign attachments (attachments by other users)

### **wagtailmedia :**

- upload
- managing video and audio files within the admin

### **django-filebrowser-no-grappelli:**

- browse directories on your server and upload/delete/edit/rename files.
- include images/documents to your models/database using the FileBrowseField.
- select images/documents with TinyMCE.
- Cannot copy/drag and drop files.

### **django-ajaximage:**

- Upload/remove files
  - Not enough features

### **pyuploadcare:**

- file uploading, storage, transformations, and delivery
- Drag & drop
- uploads for the web
- remove
- Uploading
  - Add a file uploading capability to your website or app.
  - Upload files of any type and up to 5 TB in size.
  - Get files from various upload sources, including local storage, camera, social media, and cloud storage services.
  - Upload multiple files in one go.
  - Track upload jobs with an individual progress bar for each file.
  - Speed up the uploading with the uploading network (it works like CDN).
- Image Handling
  - Show image previews.
  - Implement custom image crop options.
  - Edit, enhance, and filter images in any browser.
- Validation
  - Validate files by their format or size.
  - Automatically resize large incoming images.
- Security
  - Make your uploading system compatible with SOC2, HIPAA, and more.
  - Prevent remote code execution through File Uploading.
  - Prevent code execution in uploaded files like SVG, html and xml.
- Reliability
  - All of your uploads go to the storage covered by SLA and with a 99.9% uptime.

### **python-fsutil:**

- Upload/clean directory
- Upload/copy/create/delete/download/list/make/move/read/rename/search/write files
- Convert size string to bytes
- Extract zip files
- Get the file creation/last modified date
- Get file size
- Determine whether the file is empty

### **django-file-picker :**

- uploading, browsing, and inserting various forms of media into HTML form fields
- Cannot drag and drop files.

### **wagtail-filepreviews :**

- Upload/copy file
- Crop/view images
- Drag & drop
- Get files from various upload sources, including local storage, microphone, camera, social media, and cloud storage services.
- Output format: .png, .jpeg, .webp, .jpg
- Set preview dimensions
- Metadata includes OCR, EXIF, checksum, RAW, PSD layers, multimedia, webpage, sketch
- Cannot delete/rename files

### **django smart share :**

- Upload/ delete files
- share any directory on your local system
- Active (whether to show this directory for the allowed user or not), Show Files, Show Directories, Show Hidden Directories, Download, Upload, Delete, Create

- Cannot copy/move
- Cannot preview/edit

### **django-fileman :**

- Add/delete/rename/upload files

### **welut :**

- convert the .epub, .mobi or .pdf files as images.
- Usefull to manage e-library or e-commerce bassed on e-book system

### **django-img:**

- Upload
- Abstract base allows any number of repositories with custom configurations
- Filter declarations can travel with apps (like CSS)
- Auto-generates filtered images
- One (primary) template tag
- Only images, not any file
- Filters cannot be applied to individual images
- No effort to present filters to users, either admin or visitors
- No built-in categorization and/or tagging

### **Mayan EDMS-electronic document manager :**

- store, introspect, and categorize files, with a strong emphasis on preserving the contextual and business information of documents. It can also OCR, preview, label, sign, send, and receive those files
- cannot Get files from servers
- cannot copy

This comparison supports the idea that pyuploadcare can be used for our purpose.