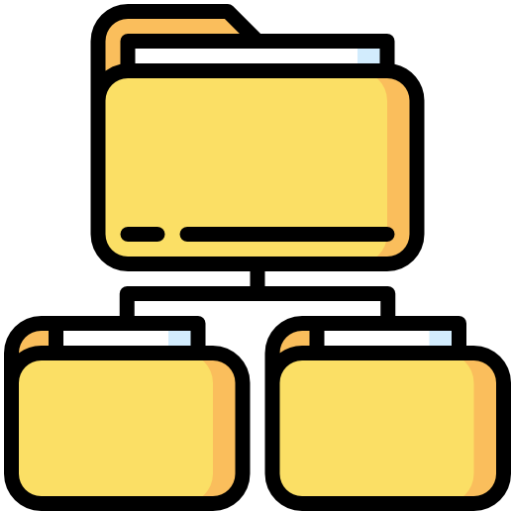

Automated File Organizer with Python

Prepared By:
Mahmoud Alsalloum

Table of Contents

Contents

Table of Contents 2
Technology Stack 3
Project Overview 3
Implementation Steps 4

Technology Stack

Ubuntu-Client – The environment where the script was developed and tested.

Python – The Programming language used for its platform independence and extensive libraries.

OS Module – Used to interact with the file system.

Shutil Module – Used for moving files between directories.

Logging Module – Implemented to track the organization process.

Systemd – Used to ensure the script runs automatically after system boot.

Project Overview

Welcome to my project!

Thank you for taking the time to explore my portfolio. Managing files manually can be tedious and error-prone, especially when dealing with a large number of files. In this project, I developed a python script to automate file organization. The script categorizes files into subfolders based on their types (like Pictures / Documents …). This solution is compatible with Windows, macOS, and Linux, making it versatile and effiecient.

Automation not only saves time but also ensures consistency and accuracy in file management.

Implementation Steps

I wrote a Python script that performs the following tasks:

- Scans a specified directory for files.
- Categorizes files based on their extensions (like .jpg, .pdf).
- Moves files into corresponding subfolders (like Pictures, Documents).
- Continuously monitors the directory for new files and organizes them automatically.

See Figure 1.

```
  GNU nano 7.2                                                    /home/mahmoud/file_organizer.py
import os
import shutil
import logging
import time


logging.basicConfig(filename='file_organizer.log', level=logging.INFO,
                    format='%(asctime)s - %(message)s')


FILE_CATEGORIES = {
    "Pictures": [".jpg", ".jpeg", ".png", ".gif"],
    "Documents": [".pdf", ".docx", ".txt"],
    "Videos": [".mp4", ".mkv", ".avi"],
    "Music": [".mp3", ".wav"],
    "Archives": [".zip", ".rar", ".tar.gz"]
}

def organize_files(directory):
    for filename in os.listdir(directory):
        file_path = os.path.join(directory, filename)


        if os.path.isdir(file_path):
            continue


        file_extension = os.path.splitext(filename)[1].lower()
        for category, extensions in FILE_CATEGORIES.items():
            if file_extension in extensions:
                category_path = os.path.join(directory, category)
                os.makedirs(category_path, exist_ok=True)
                shutil.move(file_path, os.path.join(category_path, filename))
                logging.info(f"Moved {filename} to {category}/")
                break


if __name__ == "__main__":
    target_directory = "/home/mahmoud/Downloads"
    while True:
        organize_files(target_directory)
        time.sleep(10)
```

Figure 1: Python Script for File Organization

To ensure the script runs automatically after system boot, I created a Systemd service. I created a file at /etc/systemd/system/file_organizer.service with the following content. See Figure 2.

```
  GNU nano 7.2
[Unit]
Description=Automated File Organizer
After=network.target

[Service]
ExecStart=/usr/bin/python3 /home/mahmoud/file_organizer.py
Restart=always
User=mahmoud
WorkingDirectory=/home/mahmoud

[Install]
WantedBy=multi-user.target
```

Figure 2: Systemd Service file

Then I reload the Systemd configuration. See Figure 3.

```
mahmoud@ubuntu-client:~$ sudo systemctl daemon-reload
```

Figure 3: Reload the System configuration

Then I enable start the service to start on boot. See Figure 3.

```
mahmoud@ubuntu-client:~$ sudo systemctl enable file_organizer.service
mahmoud@ubuntu-client:~$ sudo systemctl start file_organizer.service
```

Figure 4: Enable and start service

To ensure the script works as expected, I tested it with a sample directory containing various file types. I downloaded multiple files with different extensions into the Downloads folder. The script successfully created corresponding subfolders (Documents / Videos / Pictures / Music etc…). The log file provided a detailed record of all actions performed by the script, confirming that the files were organized correctly. See Figure 5.

```
mahmoud@ubuntu-client:/etc/systemd/system$ cat /home/mahmoud/file_organizer.log
2025-06-30 15:02:55,166 - Moved Project-1.pdf to Documents/
2025-06-30 15:03:35,168 - Moved Project-2.pdf to Documents/
2025-06-30 15:04:25,171 - Moved Cat_August_2010-4.jpg to Pictures/
```

Figure 5: Checking the Log file