

NeatFile User Manual

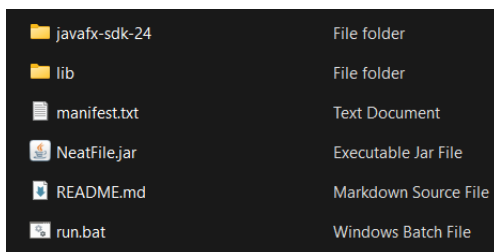
1. Introduction

NeatFile is a JavaFX-based desktop application designed to organize files automatically based on user-defined rules. It monitors selected folders and moves matching files to target directories using criteria like file name, content, extension, or last modified date.

2. Installation

1. Make sure you have Java 17 or later installed.
2. Download the latest release ZIP file from the GitHub repository - [download here](#).
3. Extract the contents of the ZIP file to a convenient location.

The image below shows everything needed for the app to run.



4. Double-click the 'run.bat' file to launch the app on Windows. If there is a pop-up saying "Windows protected your PC", click "run anyways".
5. The main window will appear with three sections: Watch Paths, Rules, and Target Path.

3. Interface Overview - [UI Preview](#)

The interface has three main sections:

- Watch Paths: Select folders to monitor for files.
- Rules: Define how files should be filtered or categorized.
- Target Path: Choose where to move matched files.
- Group dropdown: Lets you manage different sets of rules and folders.
- Finalize button: Saves your current configuration to "groups.json" file.

4. How to Use

1. Click the '+' button next to the group dropdown to create a new group.
2. Click the '+' under Watch Paths to add folders to monitor.
3. Click the folder icon under Target Path to choose where matching files go.
4. Add rules using the buttons in the Rules section:
 - Extension Rule

- Name Rule
- String Rule
- Category Rule
- Last Modified Rule

5. Remove rules and watch paths by clicking on them, a warning box will appear.
7. Click 'Finalize' to save your configuration.
8. The app will automatically scan every 5 seconds.

5. Rules Explained

- **Extension Rule:** Matches files by extension (e.g., .pdf, .txt).

Example:

This will move any files ending in .pdf, or .docx. to the target directory. Exclude the '.'.

Name Rule

Matches keywords in file names. (can check if you're using case-sensitive or using regex)

Example:

This will look for any files beginning with **4 digits**, followed by an **underscore**, then any number of any characters, ending in **.pdf**.

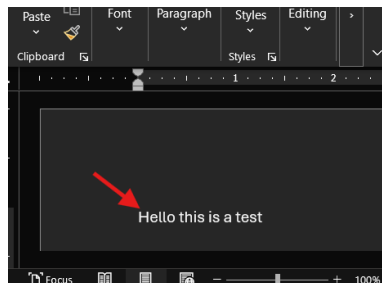
Files Moved: 2022_resume.pdf, 2023_resume.pdf, 2024_resume.pdf,

Not Moved: resume_2025.pdf

String Rule

Searches for specific text inside file content. (can check if you're using case-sensitive or regex)

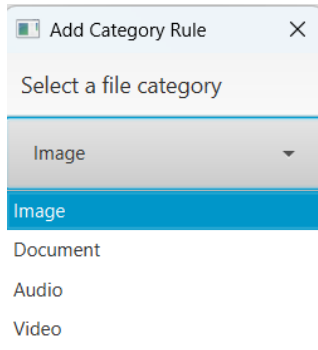
Example:



This word document has the word “Hello” in it. The file will be moved since “hello” in the String Rule is **not** marked as ‘Case Sensitive.’

Category Rule

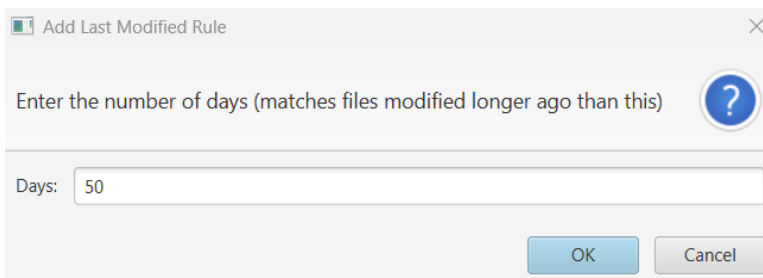
Groups files by general type (document, image, video, etc).



Example: When 'Image' is selected, any files ending in .png, .jpg, .jpeg, .gif, and so on will be moved.

Last Modified Rule

Matches files modified more than 'X' days ago.

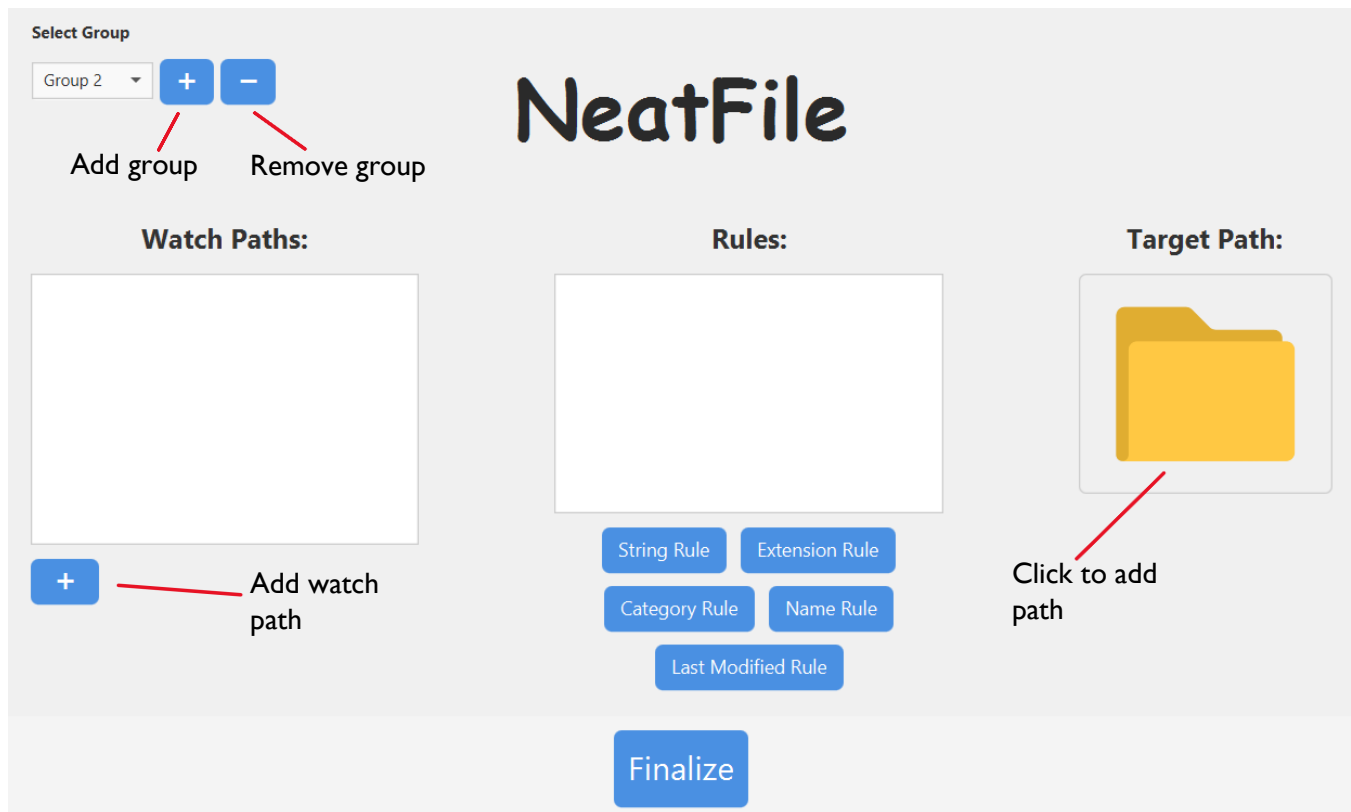


Example: This will move any files last modified over 50 days ago or more.

6. Known Issues

- Must click 'finalize' button to save any changes.
- Deleting a group does not remove it from groups.json automatically – it must be finalized to save.
- A conflicting group will not remove itself from the UI, the user will have to remove it by clicking the '-' button on the selected group.

UI Preview



8. FAQ

Q: What Java version do I need?

A: Java 17 or newer.

Q: Can I use this on Mac or Linux?

A: Only Windows is officially supported at the moment.

Q: What happens if a file with the same name exists in the target folder?

A: The new file is renamed to avoid overwriting.

Q: Can I edit the rules after they're added?

A: Once a rule is added, it **cannot** be edited, it can only be removed.

Q: Will it run in the background while the app is closed?

A: No, unfortunately the app needs to be running to actively scan for files.

Q: Is a log saved anywhere?

A: You can only view live changes via the terminal, and you can view configurations in the groups.json file.