

# 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.
3. Extract the contents of the ZIP file to a convenient location.
4. Double-click the 'run.bat' file to launch the app on Windows.

## 3. Interface Overview

The interface has three main sections:

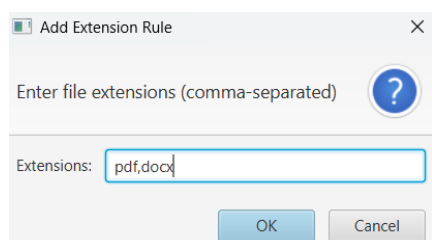
- Watch Paths: Select folders to monitor for files.
- Rules: Define how files should be filtered or categorized.
- Target Path: Choose where to move matched files.
- Group dropdown: Lets you manage different sets of rules and folders.
- Finalize button: Saves your current configuration to "groups.json" 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.
5. Click 'Finalize' to save your configuration.
6. The app will automatically scan every 5 seconds.

## 5. Rules Explained

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

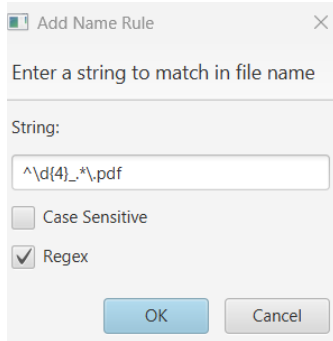


**Example:**

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)



The 'Add Name Rule' dialog box has a title bar with a close button. It contains a label 'Enter a string to match in file name' above a text input field. The input field contains the regex string '^\\d{4}\_.\*\\.pdf'. Below the input field are two checkboxes: 'Case Sensitive' (unchecked) and 'Regex' (checked). At the bottom are 'OK' and 'Cancel' buttons.

### **Example:**

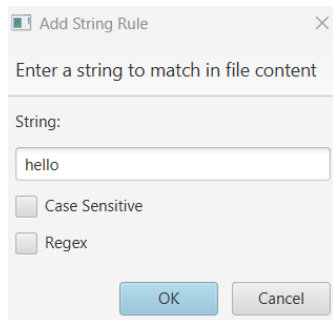
This will look for any files beginning with **4 digits**, followed by an **underscore**, then any number of any characters, ending in **.pdf**.

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

Not Moved: resume\_2025.pdf

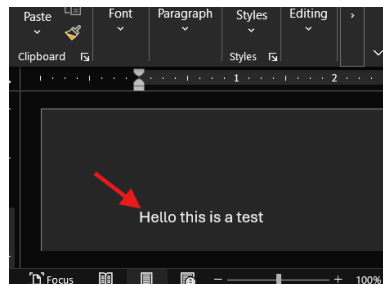
## String Rule

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



The 'Add String Rule' dialog box has a title bar with a close button. It contains a label 'Enter a string to match in file content' above a text input field. The input field contains the string 'hello'. Below the input field are two checkboxes: 'Case Sensitive' (unchecked) and 'Regex' (unchecked). At the bottom are 'OK' and 'Cancel' buttons.

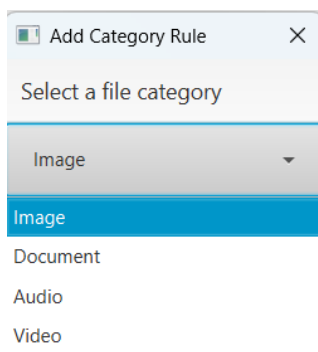
### **Example:**



**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).

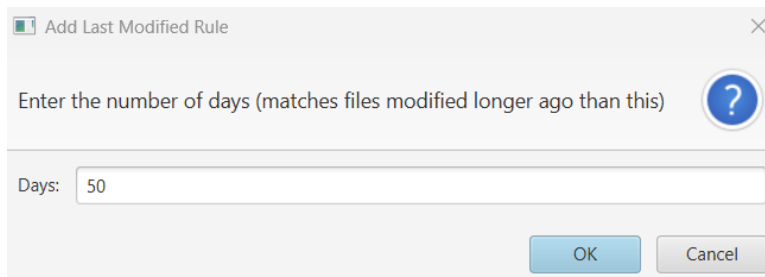


The 'Add Category Rule' dialog box has a title bar with a close button. It contains a label 'Select a file category' above a dropdown menu. The dropdown menu is open, showing a list of categories: 'Image' (selected), 'Document', 'Audio', and 'Video'.

**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.



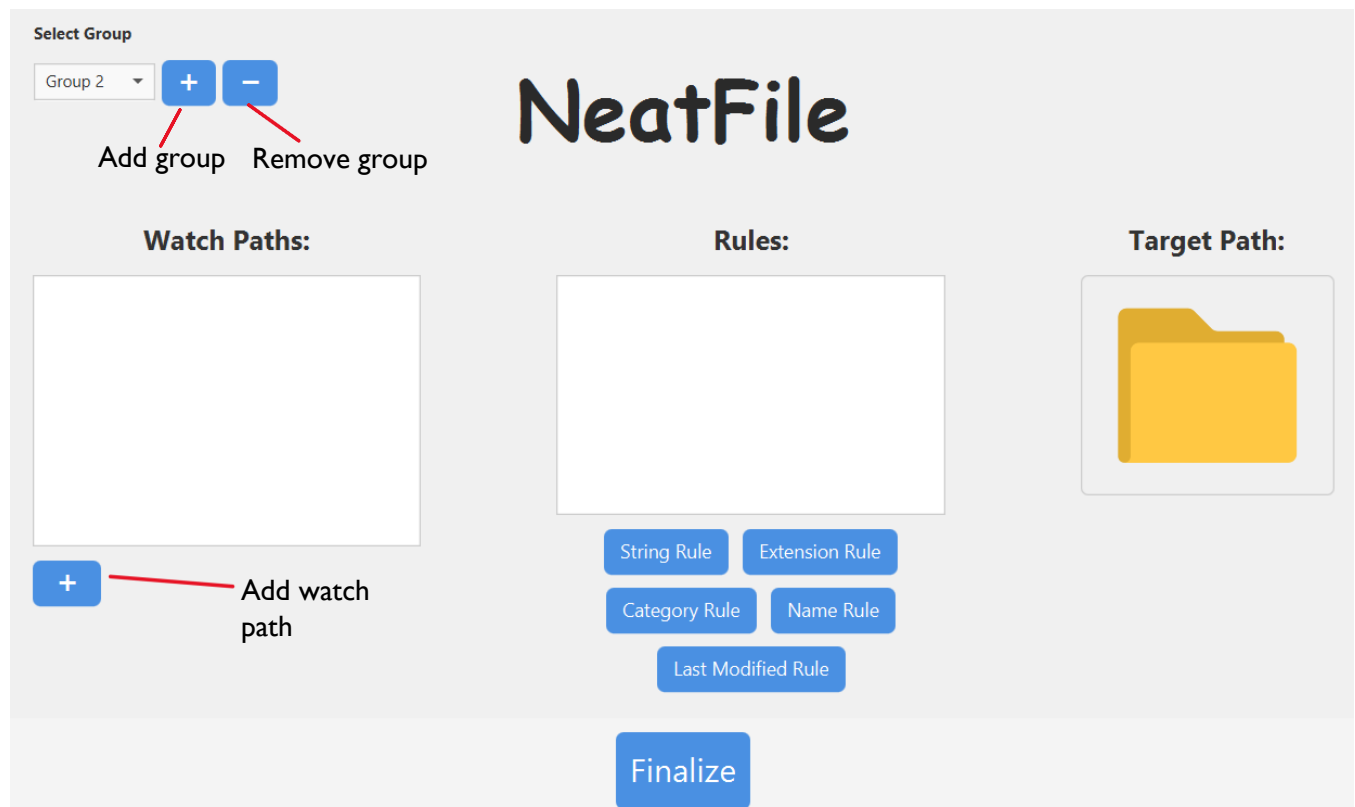
A dialog box titled "Add Last Modified Rule" with a close button (X) in the top right corner. It contains a text input field with the placeholder "Enter the number of days (matches files modified longer ago than this)" and a question mark icon. Below the input field, the text "Days:" is followed by a text box containing the number "50". At the bottom right, there are two buttons: "OK" and "Cancel".

**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.
- Rules **cannot** be edited, only added and removed.

## UI Preview



The NeatFile UI preview shows a main interface with the title "NeatFile" in the center. On the top left, there is a "Select Group" section with a dropdown menu showing "Group 2" and two buttons: a blue "+" button labeled "Add group" and a blue "-" button labeled "Remove group". Below this, there are three main sections: "Watch Paths:", "Rules:", and "Target Path:". The "Watch Paths:" section has a large empty box and a blue "+" button labeled "Add watch path". The "Rules:" section has a large empty box and five buttons: "String Rule", "Extension Rule", "Category Rule", "Name Rule", and "Last Modified Rule". The "Target Path:" section has a yellow folder icon. At the bottom center, there is a large blue "Finalize" button.

## 7. 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: 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.