St. JOSEPH'S
GROUP OF INSTITUTIONS
OMR, CHENNAI - 119

Placement Empowerment Program

Cloud Computing and DevOps Centre

Create a script that backs up your entire Git repository to a local folder daily

Name: Esly Abro **Department: IT**

St. JOSEPH'S COLLEGE OF ENGINEERING | St. JOSEPH'S INSTITUTE OF TECHNOLOGY
AUTONOMOUS INSTITUTIONS, AFFILIATED TO ANNA UNIVERSITY

Automated Git Backup System

The task was to create a system that automatically backs up a local Git repository to a remote GitHub repository. The script ensures that whenever changes are made in the local repository, they are committed and pushed to the backup repository on GitHub. If no changes are detected, the script outputs "up to date" to indicate that the local repository is synchronized with the backup repository.

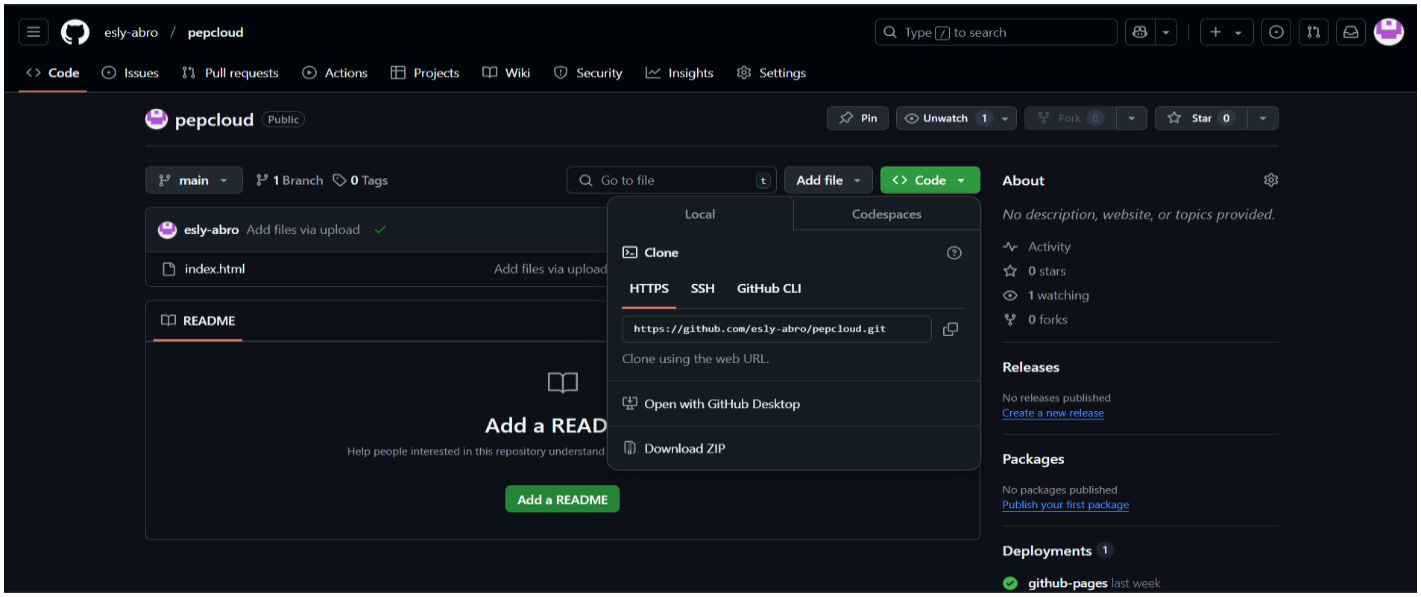
Objectives:

- **Automate Backup:** Set up a process that automatically pushes any changes from a local Git repository to a backup GitHub repository.
- **Ensure Version Control:** Maintain a separate backup of your project on GitHub to reduce the risk of data loss.
- **Simplify Regular Backups:** Automate the backup process so no manual intervention is needed.
- **Maintain Sync:** Make sure the backup repository is always up-to-date with the latest changes from the local repository.
- **Increase Data Security:** Regularly back up your code to GitHub, ensuring it is stored safely and securely.

Step-by-Step Overview:

Step 1: Set Up Your Local Git Repository

1. **Clone Your Git Repository:** Clone your existing Git repository to your desktop if it's not already cloned. Use the following command: `git clone https://github.com/your-username/your-repository.git`



```
https://github.com/esly-abro/pepcloud.git
```

Clone using the web URL.

Step 2: Create a Backup Repository on GitHub

1. Go to GitHub and create a new repository (e.g., `backup-your-repository-name`).
2. Choose whether the repository is **public** or **private**, depending on your needs.
3. **Do not initialize the repository** with files like README, LICENSE, etc.

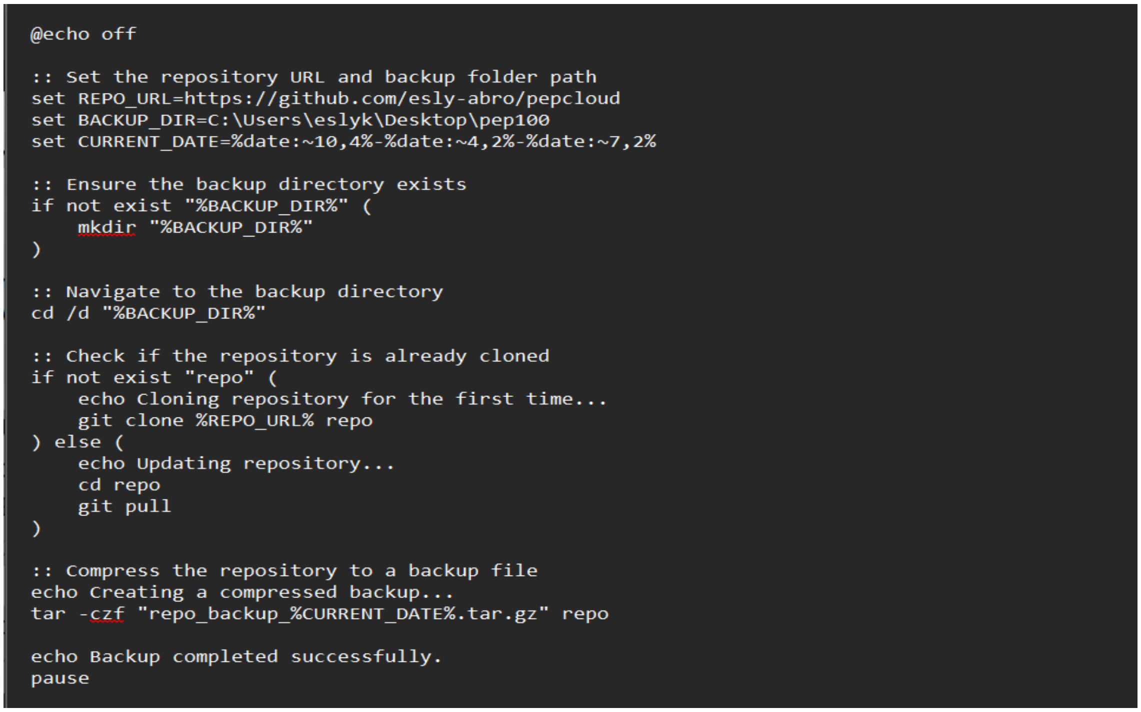
Step 3: Write the Backup Script

Create a batch script (e.g., `backup_git_repo.bat`) that performs the following:

- Check if there are changes in the local repository.
- Commit and push changes to the backup repository if new changes are found.
- If there are no changes, the script will output “up to date.”

Example of the script:

```
@echo off
cd path\to\your\local\repository
git add .
git commit -m "Automated backup commit"
git push https://github.com/your-username/backup-your-repository-name.git master
if %errorlevel% neq 0 (
    echo "Up to date, no new changes to be made."
) else (
    echo "Backup successful."
)
pause
```



```
@echo off

:: Set the repository URL and backup folder path
set REPO_URL=https://github.com/esly-abro/pepcloud
set BACKUP_DIR=C:\Users\eslyk\Desktop\pep100
set CURRENT_DATE=%date:~10,4%-%date:~4,2%-%date:~7,2%

:: Ensure the backup directory exists
if not exist "%BACKUP_DIR%" (
    mkdir "%BACKUP_DIR%"
)

:: Navigate to the backup directory
cd /d "%BACKUP_DIR%"

:: Check if the repository is already cloned
if not exist "repo" (
    echo Cloning repository for the first time...
    git clone %REPO_URL% repo
) else (
    echo Updating repository...
    cd repo
    git pull
)

:: Compress the repository to a backup file
echo Creating a compressed backup...
tar -czf "repo_backup_%CURRENT_DATE%.tar.gz" repo

echo Backup completed successfully.
pause
```

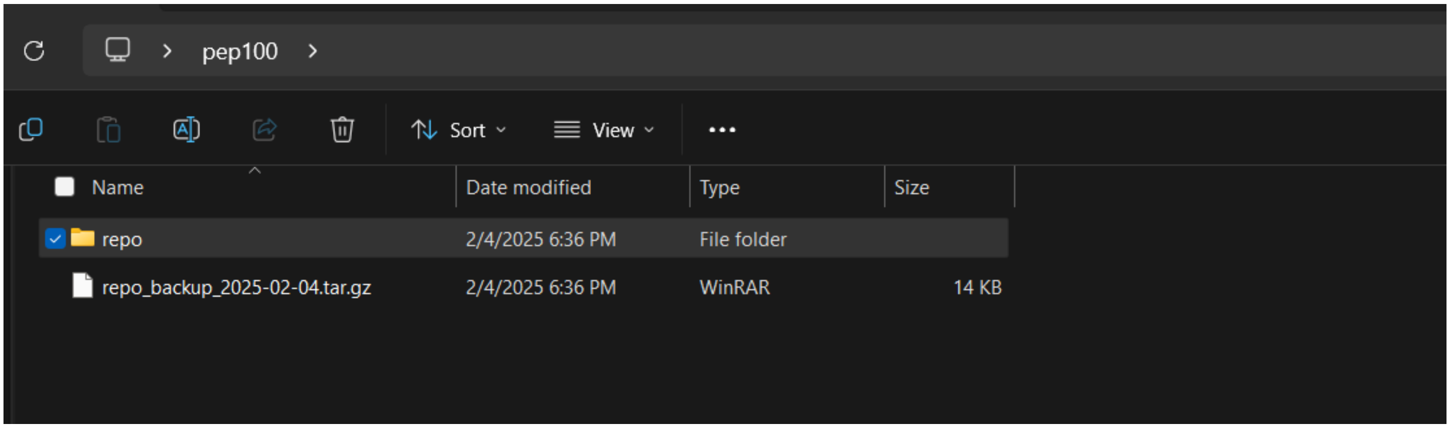
Step 4: Save and Run the Script

1. Open **Notepad** and paste the script.
2. Save the file as `backup_git_repo.bat` on your desktop or another location.
3. Run the script:

a. If no changes have been made since the last backup, the script will output "up to date, no new changes to be made."
b. If new changes are detected, the script will automatically commit and push those changes to the backup repository on GitHub.

Step 5: Verify the Backup

1. After running the script, check the backup repository on GitHub to confirm that it has been updated with the latest changes.
2. If new changes were made, they should now be visible in the backup repository.



Name	Date modified	Type	Size
☑ repo	2/4/2025 6:36 PM	File folder	
repo_backup_2025-02-04.tar.gz	2/4/2025 6:36 PM	WinRAR	14 KB

Task Outcomes:

1. **Automated Backup Process:**

Successfully set up a system that automatically pushes changes from the local Git repository to a remote GitHub backup repository. This eliminates the need for manual backups, ensuring your project remains backed up regularly.

2. **Up-to-Date Backup:**

The backup repository on GitHub is always kept up-to-date, with all the changes made to the local repository. Whenever you make updates to the local repository, they will be reflected in the backup repository.

3. **Improved Data Security:**

By regularly backing up the repository to GitHub, an additional layer of data security is provided. This reduces the risk of data loss, as your project is safely stored in the cloud.

4. **Time-Saving:**

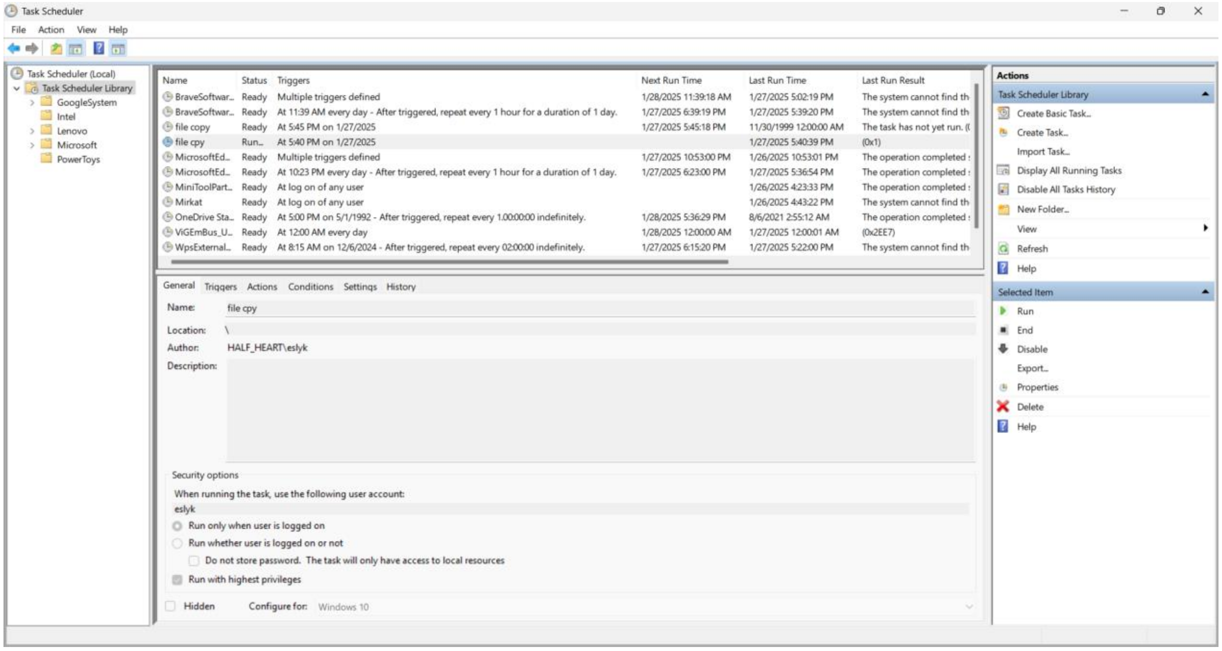
The automated backup process eliminates the need to manually commit and push changes to the backup repository. This saves time and ensures that your backup is always current without needing any extra effort.

Task Scheduler: Automating Daily Backups

To ensure the backup process runs daily without manual intervention, you can automate the task using the Windows Task Scheduler.

Steps to Set Up a Daily Backup Task:

1. **Open Task Scheduler:**
   a. Press `Win + R`, type `taskschd.msc`, and press Enter to open Task Scheduler.



Name	Status	Triggers	Next Run Time	Last Run Time	Last Run Result
BraveSoftwar...	Ready	Multiple triggers defined	1/28/2025 11:39:18 AM	1/27/2025 5:02:19 PM	The system cannot find th
BraveSoftwar...	Ready	At 11:39 AM every day - After triggered, repeat every 1 hour for a duration of 1 day.	1/27/2025 6:39:19 PM	1/27/2025 5:39:20 PM	The system cannot find th
file copy	Ready	At 5:45 PM on 1/27/2025	1/27/2025 5:45:18 PM	11/30/1999 12:00:00 AM	The task has not yet run. ((
file cpy	Run...	At 5:40 PM on 1/27/2025		1/27/2025 5:40:39 PM	(0x1)
MicrosoftEd...	Ready	Multiple triggers defined	1/27/2025 10:53:00 PM	1/26/2025 10:53:01 PM	The operation completed :
MicrosoftEd...	Ready	At 10:23 PM every day - After triggered, repeat every 1 hour for a duration of 1 day.	1/27/2025 6:23:00 PM	1/27/2025 5:36:54 PM	The operation completed :
MiniToolPart...	Ready	At log on of any user		1/26/2025 4:23:33 PM	The operation completed :
Mirkat	Ready	At log on of any user		1/26/2025 4:43:22 PM	The system cannot find th
OneDrive Sta...	Ready	At 5:00 PM on 5/1/1992 - After triggered, repeat every 1.00:00:00 indefinitely.	1/28/2025 5:36:29 PM	8/6/2021 2:55:12 AM	The operation completed :
ViGEmBus_U...	Ready	At 12:00 AM every day	1/28/2025 12:00:00 AM	1/27/2025 12:00:01 AM	(0x2EE7)
WpsExternal...	Ready	At 8:15 AM on 12/6/2024 - After triggered, repeat every 02:00:00 indefinitely.	1/27/2025 6:15:20 PM	1/27/2025 5:22:00 PM	The system cannot find th

General | Triggers | Actions | Conditions | Settings | History

Name: file cpy

Location: \

Author: HALF_HEART\eslyk

Description:

Security options

When running the task, use the following user account:

eslyk

● Run only when user is logged on

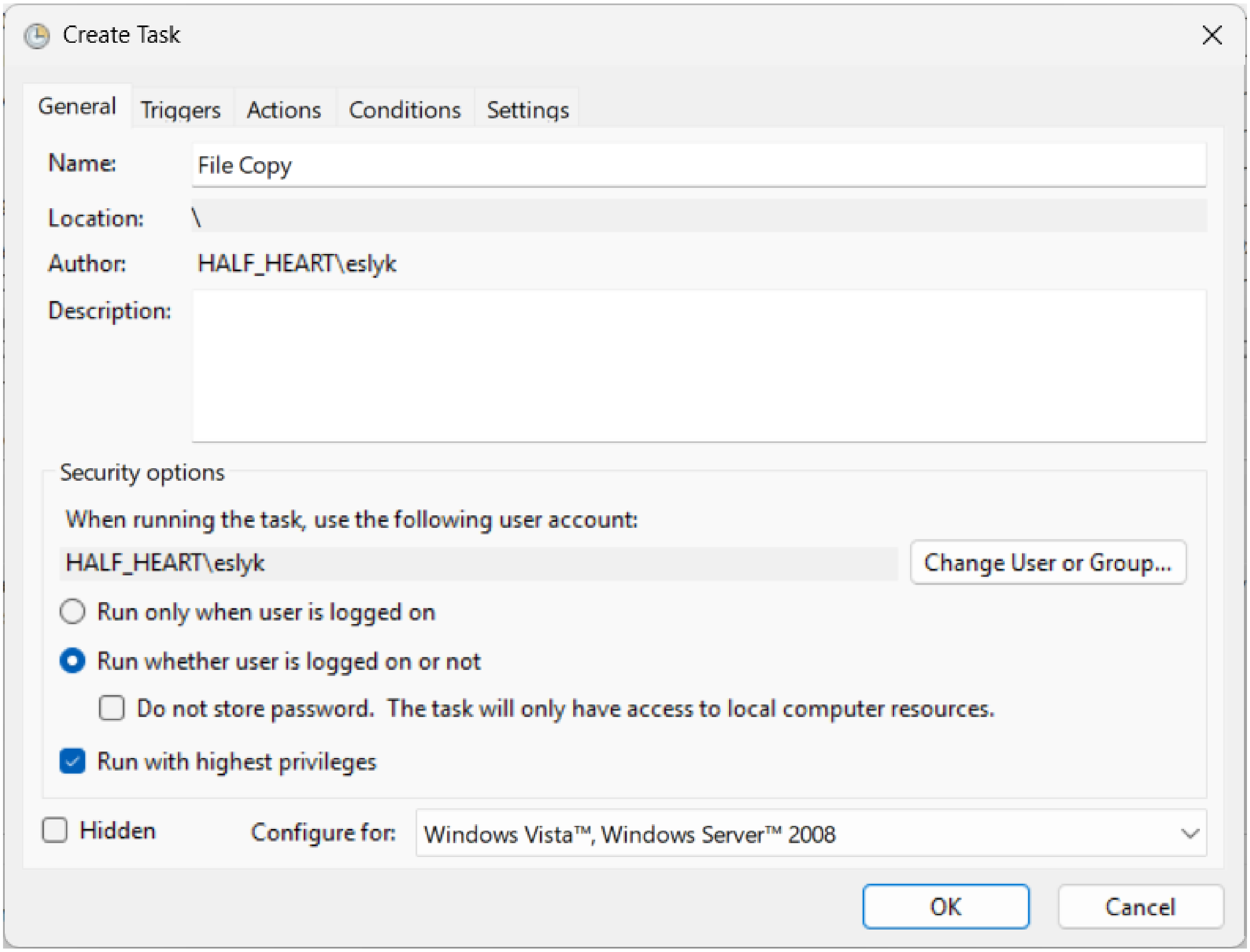
○ Run whether user is logged on or not

□ Do not store password. The task will only have access to local resources

☑ Run with highest privileges

□ Hidden   Configure for: Windows 10

2. **Create a New Task:**
   a. In Task Scheduler, click **Create Basic Task**.
   b. Name the task "Daily Git Backup" and provide a description (optional).



Name: File Copy

Location: \

Author: HALF_HEART\eslyk

Description:

Security options

When running the task, use the following user account:

HALF_HEART\eslyk

○ Run only when user is logged on

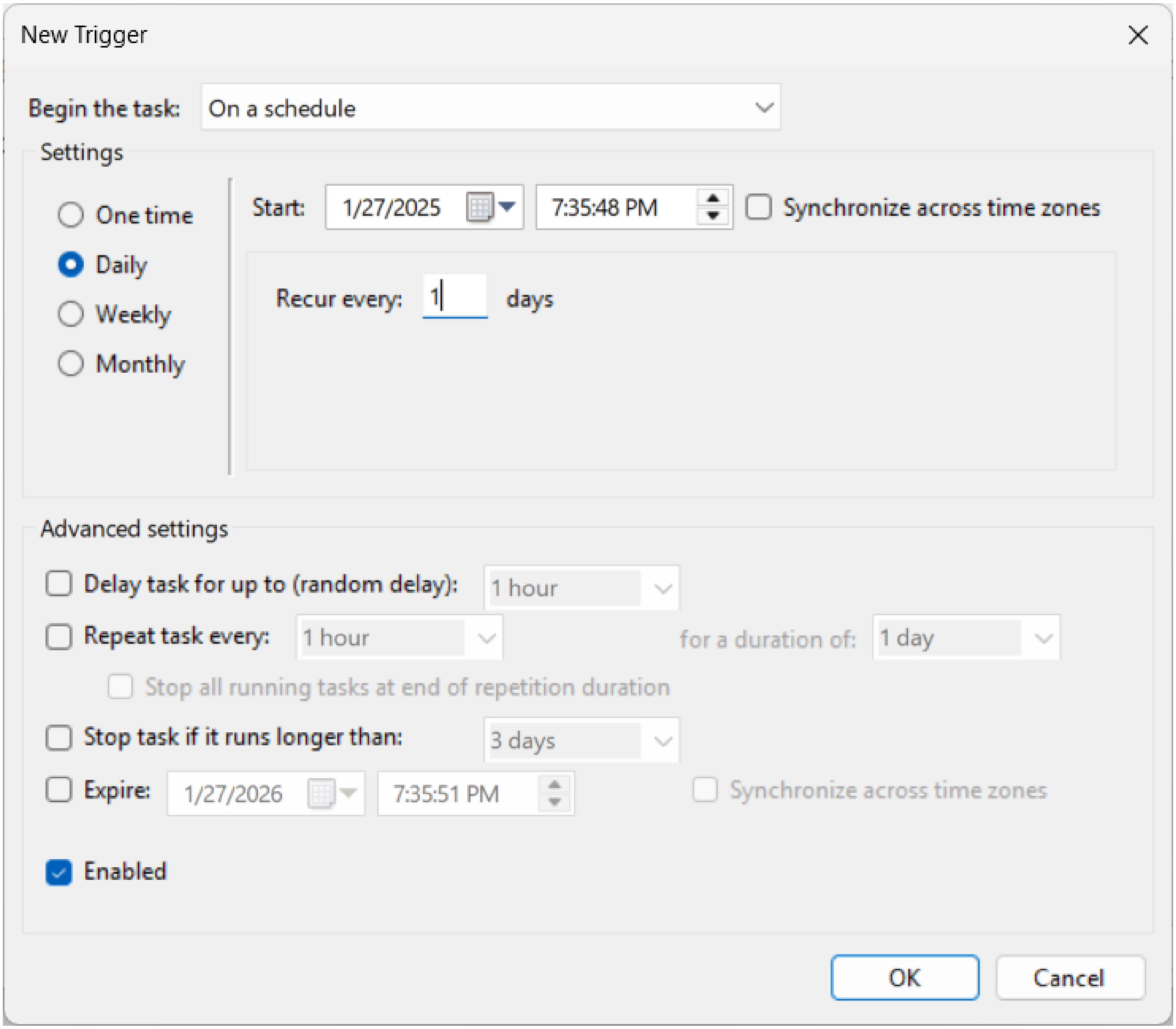
● Run whether user is logged on or not

□ Do not store password. The task will only have access to local computer resources.

☑ Run with highest privileges

□ Hidden   Configure for: Windows Vista™, Windows Server™ 2008

3. **Set the Trigger:**
   a. Select **Daily** and set the time you want the task to run (e.g., 12:00 AM).



New Trigger

Begin the task: On a schedule

Settings

○ One time
● Daily
○ Weekly
○ Monthly

Start: 1/27/2025 7:35:48 PM □ Synchronize across time zones

Recur every: 1 days

Advanced settings

□ Delay task for up to (random delay): 1 hour

□ Repeat task every: 1 hour for a duration of: 1 day

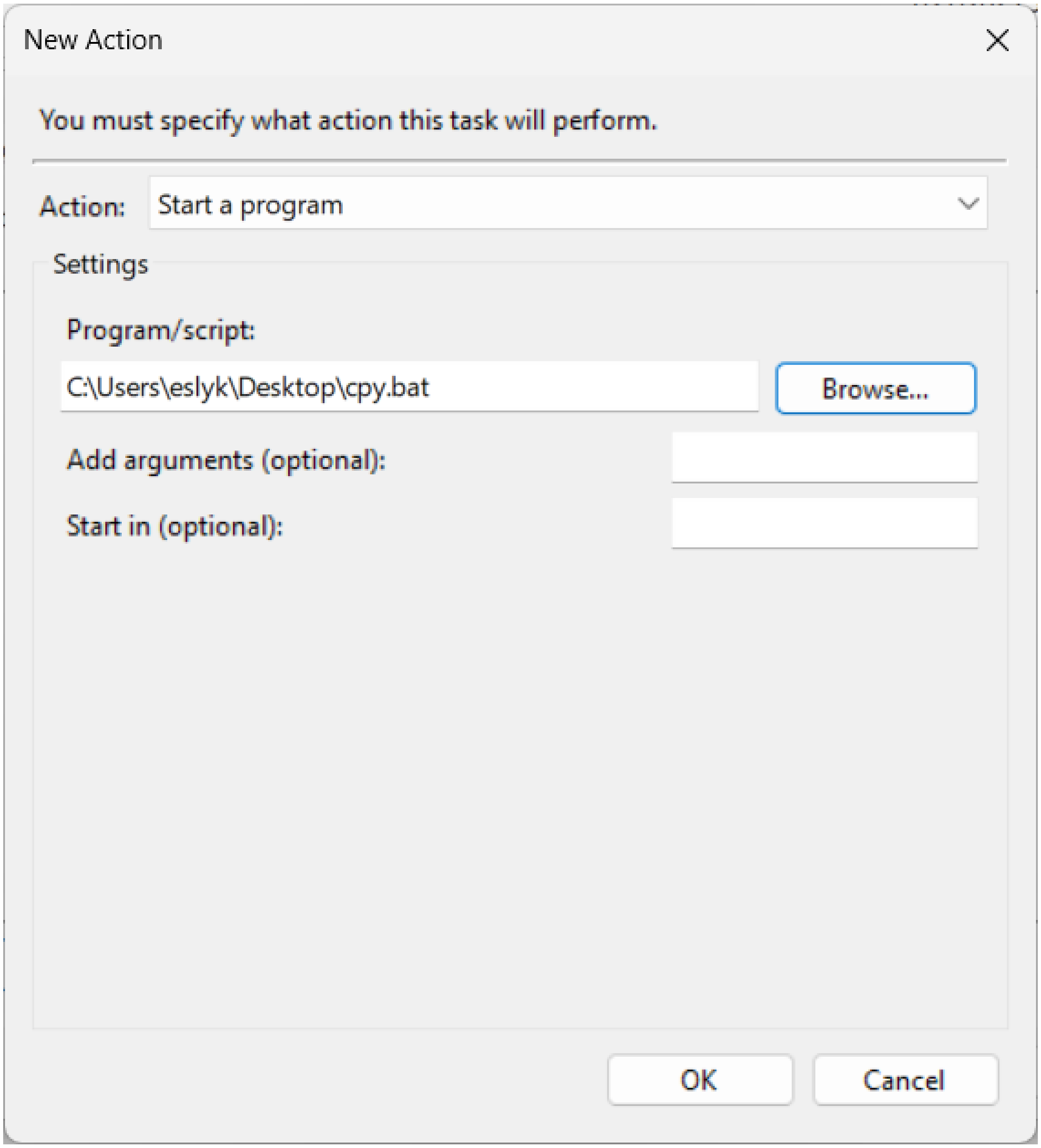
□ Stop all running tasks at end of repetition duration

□ Stop task if it runs longer than: 3 days

□ Expire: 1/27/2026 7:35:51 PM □ Synchronize across time zones

☑ Enabled

4. **Set the Action:**
    a. Choose **Start a Program** and browse to the location of the `backup_git_repo.bat` script you created.



New Action

You must specify what action this task will perform.

Action: Start a program

Settings

Program/script:

C:\Users\eslyk\Desktop\cpy.bat

5. **Finish the Task:**
   a. Review the settings and click **Finish** to schedule the task.
   b. Ensure that the task is set to run even if the computer is on battery power, if necessary.

Conclusion:

This system guarantees that your Git repository is backed up regularly, with all changes automatically pushed to the backup repository. The Task Scheduler ensures that backups happen daily without your intervention. This provides enhanced data security, simplifies the backup process, and saves you time, ensuring your project is always safe and up-to-date.