



Gizmo Gurus LLC

Github Auto Backup

User manual for Github Auto Backup PC application

Copyright Statement

This manual, including all content, illustrations, instructions, and the software described herein (the "Software"), is the sole and exclusive intellectual property of **Gizmo Gurus LLC**. All rights are reserved under applicable copyright, trademark, and intellectual property laws. Any unauthorized duplication, distribution, modification, reverse engineering, decompilation, or use of this material or the Software is strictly prohibited and may result in severe civil and criminal penalties.

Gizmo Gurus LLC grants a limited, non-transferable, non-exclusive license to use the Software solely in accordance with the terms outlined in this manual. No part of this manual or the Software may be reproduced or transmitted in any form, including electronic, mechanical, photocopying, recording, or any other means, without prior written permission from **Gizmo Gurus LLC**.

The Software is provided "as is," without warranty of any kind, express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, and non-infringement. **Gizmo Gurus LLC** reserves the right to make changes to the Software or this manual at any time without notice.

For inquiries, permissions, or licensing agreements, please contact: **Gizmo Gurus LLC**

Contents

Copyright Statement.....	1
1. Introduction.....	3
2. PC Requirements.....	3
3. Installation.....	3
4. User Interface Overview.....	4
5. Application Use.....	4
6. Additional Features.....	5
7. Logging.....	6
8. Troubleshooting.....	6
Disclaimer.....	7

1. Introduction

GitHub Auto Backup is a Windows application designed to automate the process of backing up GitHub repositories. The application allows users to set backup intervals, specify repository locations, and manage backup schedules. It runs in the background and can be accessed via a system tray icon.

2. PC Requirements

- **Operating System:** Windows 10 or later
 - **.NET Runtime:** .NET 8.0
 - **Processor:** 1.6 GHz or faster
 - **RAM:** 8 GB or more
 - **Disk Space:** 100 MB for application files and additional space for logs
-

3. Installation

- 3.1. Download the Installer:** Obtain the installer for GitHub Auto Backup.
 - 3.2. Run the Installer:** Execute “Github Auto Backup Setup.exe” and follow the on-screen instructions.
 - 3.2.1. Choose Install Location:** Select the location where you want to install the application.
 - 3.2.2. Create Shortcuts:** Choose whether to create desktop and Start menu shortcuts.
 - 3.2.3. Auto-Start on Login:** The installer will update the Windows registry to ensure the application starts automatically when you log in to Windows.
 - 3.3. Launch the Application:** After installation, launch the application from the Start menu or desktop shortcut.
 - 3.4. Remove Application:** The uninstaller will remove the application, shortcuts, and registry entries.
-

4. User Interface Overview

Main Window

- **Folder where Repositories are stored:** A text box where you can specify the location of your GitHub repository.
- **Select Folder:** A button to open a folder browser dialog to select the repository location.
- **Backup Interval:** A combo box to select the backup interval (e.g., Daily, Weekly, Monthly).
- **Preferred Backup Time:** A date-time picker to set the time of day for the backup.
- **Open Log File:** A button to open the log file that records backup activities.
- **Close:** A button to hide the application window.
- **Run Backup Now:** A button to manually start the backup process.
- **Last Backup:** A text box displaying the last backup time.
- **Next Backup:** A text box displaying the next scheduled backup time.
- **Backup Status:** A text box displaying the current backup status.

5. Application Use

- **Install the Application:** Ensure that your PC meets the minimum requirements and install the application.
- **Open the Application:** Launch the application. The main window will appear.
- **Set Repository Location:** Use the "Select Folder" button to select the folder containing your GitHub repository.
- **Set Backup Interval:** Choose a backup interval from the combo box (e.g., Daily, Weekly, Monthly).
- **Set Backup Time:** Use the date-time picker to set the time of day for the backup.
- **Start Backup:** Click the "Run Backup" button to start the backup process manually or wait for the scheduled time.
- **View Logs:** Click the "Open Log" button to view the log file containing details of backup activities.
- **Hide Application:** Click the "Close" button to hide the application window. The application will continue to run in the background.
- **Access via System Tray:** Use the system tray icon to access the application quickly. Double-click the icon to open the main window or right-click the icon to open the context menu for quick actions.

5.1. Branch Existence Check

When processing a repository, the application checks if the current branch exists in the remote repository. This is done using the `RemoteBranchExistsAsync` method, which runs the Git command `ls-remote --heads origin <branchName>` to check for the branch's existence.

5.2. Warning for Non-Existent Branch

If the branch does not exist in the remote repository, the application displays a message box asking the user if they want to create the branch. If the user chooses "Yes", the branch is created in the remote repository using the Git command `push -u origin <branchName>`.

5.3. Avoiding Commits and Pushes to Main or Master

The application is designed to avoid committing or pushing changes to the main or master branches. If the current branch is main or master, the application logs a warning and sets the backup status to "Completed: Warning", skipping the commit and push steps.

6. Additional Features

6.1. System Tray Icon

The system tray icon that provides quick access to the application.

- **Double-Click:** Opens the main window of the application.
- **Right-Click:** Opens a context menu with the following options:
- **Run Backup:** Starts the backup process manually.
- **Last Backup:** Displays the last backup time.
- **Next Backup:** Displays the next scheduled backup time.
- **Settings:** Opens the main window of the application.
- **Exit:** Exits the application.

6.2. Background Processes

The backup process is initiated either manually by clicking the "Run Backup" button or automatically based on the scheduled interval. The process involves the following steps:

- **LogAction:** Logs the start of the backup process.
- **UpdateBackupStatus:** Updates the backup status to "Running".
- **Repository Check:** Checks if the specified repository location is a valid Git repository.
- **ProcessRepository:** Processes the repository by adding changes, committing them, and pushing them to the remote repository.
- **Update Next Backup Time:** Calculates and updates the next scheduled backup time.
- **LogAction:** Logs the completion of the backup process.
- **UpdateBackupStatus:** Updates the backup status to "Completed: Success" or "Completed: Error" based on the outcome.

6.3. Timer: The application uses a timer to periodically check if it's time to run the backup process. If the current time is past the next scheduled backup time, the backup process is initiated.

7. Logging

All significant actions and errors are logged to a file named "Github Auto Backup.log" located in the application's base directory. The log file can be opened by clicking the "Open Log" button.

8. Troubleshooting

Troubleshooting information

Issue	Solution
.NET 8.0 is not installed	Download and install .NET 8.0 from the official Microsoft website.
Application does not start on Windows login	Ensure the registry entry for auto-start is correctly set. Reinstall the application if necessary.
Backup process fails	Check the log file for error messages. Ensure the repository location is correct and accessible.
Repository branch does not exist in remote	Follow the prompt to create the branch in the remote repository.
Application window does not open	Right-click the system tray icon and select "Settings".
Log file does not open	Ensure the log file exists in the application's base directory. Check for file permissions.
Backup status shows "Completed: Error"	Review the log file for detailed error messages and take corrective actions.
Changes not committed or pushed	Ensure you are not on the <u>main</u> or <u>master</u> branch. Switch to a different branch and try again.

For persistent issues, contact Blum-Novotest support with the generated log files.

Disclaimer

Gizmo Gurus LLC disclaims all responsibility for bugs, errors, or issues that may arise while using this software. Users are encouraged to report issues for resolution, but **Gizmo Gurus LLC** is not liable for any damage or loss caused directly or indirectly by the software's use.

Users are responsible for ensuring that their systems meet the software's minimum requirements as outlined in this manual. Failure to comply with these requirements may result in improper functionality, errors, or reduced performance.

While **Gizmo Gurus LLC** encourages users to report any issues for resolution, the company is not liable for any damage, data loss, system interruptions, or other adverse effects caused directly or indirectly by the use of this software.