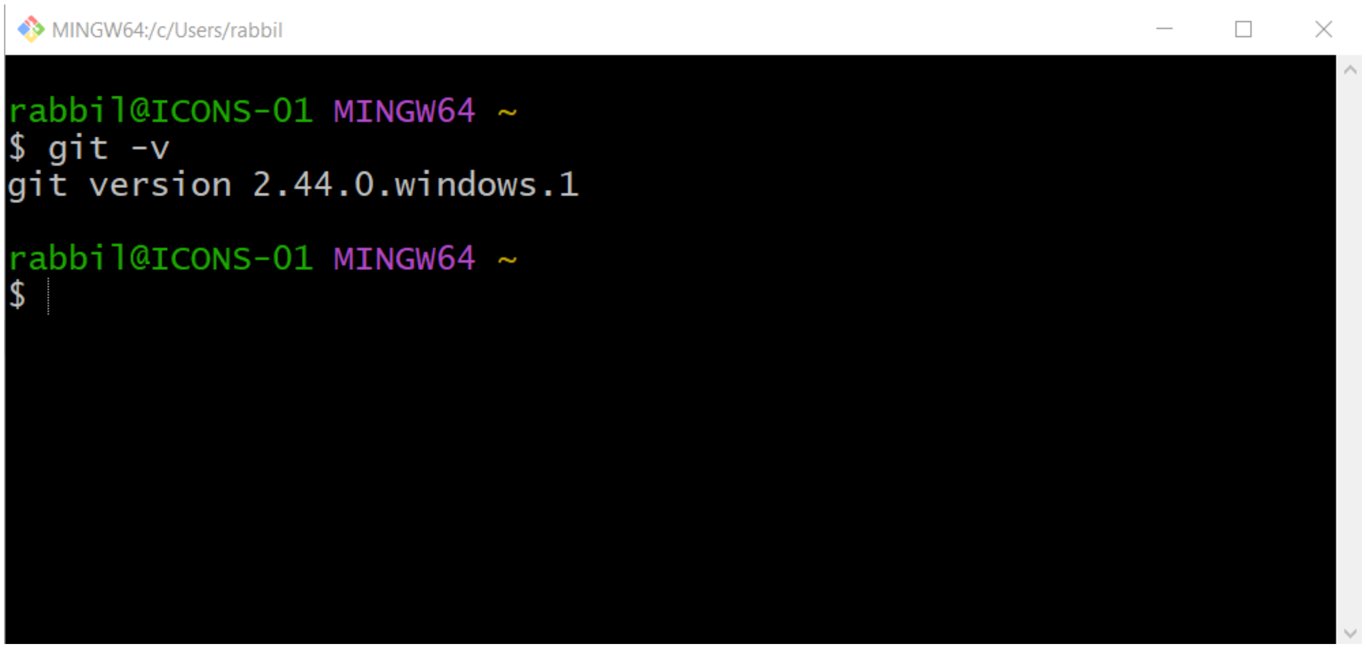
Git comes with built-in GUI tools like **git bash**, **git-gui**, and **gitk** for committing and browsing.

GitBash

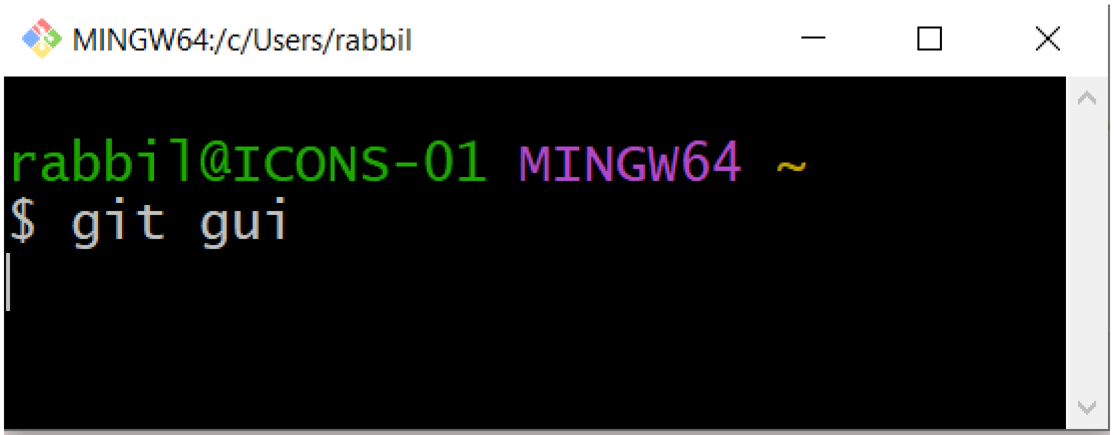
Git Bash is like a special tool for Windows computers. It lets you use Git commands just like you would on other systems. It makes Git work smoothly on Windows.



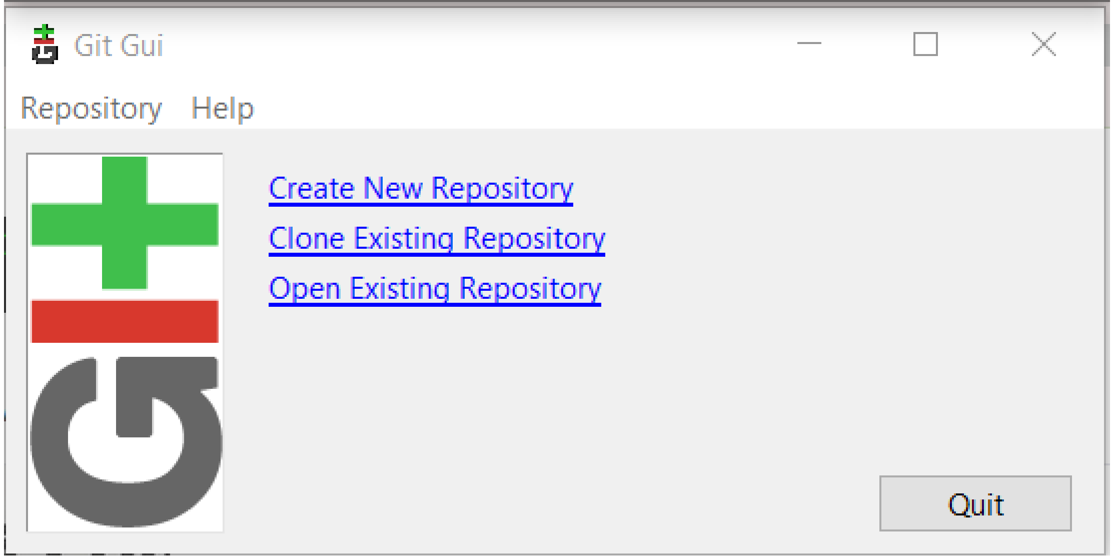
```
rabbil@ICONS-01 MINGW64 ~
$ git -v
git version 2.44.0.windows.1

rabbil@ICONS-01 MINGW64 ~
$
```

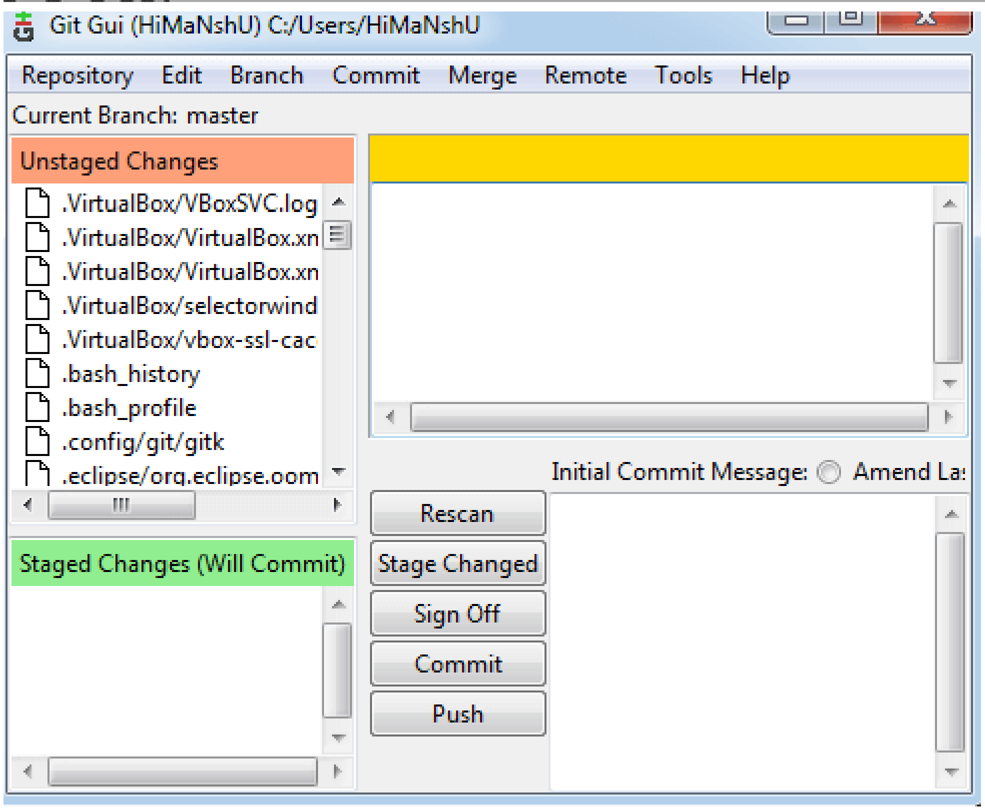
GitGui

Git GUI is a handy alternative to Git Bash. It gives you a graphical version of Git commands and includes helpful visual diff tools. Accessing it is as easy as right-clicking on a folder or location in Windows Explorer.



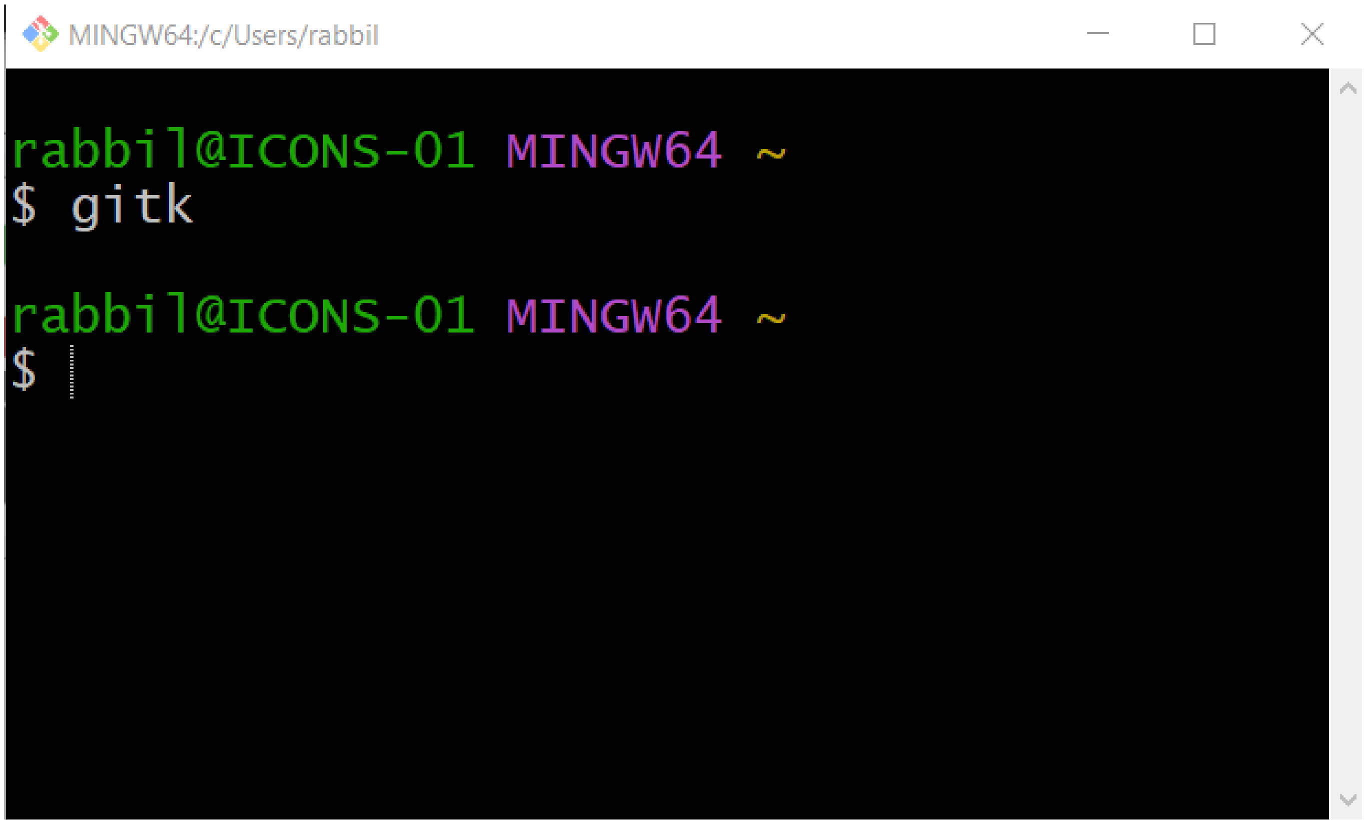
```
rabbil@ICONS-01 MINGW64 ~
$ git gui
```



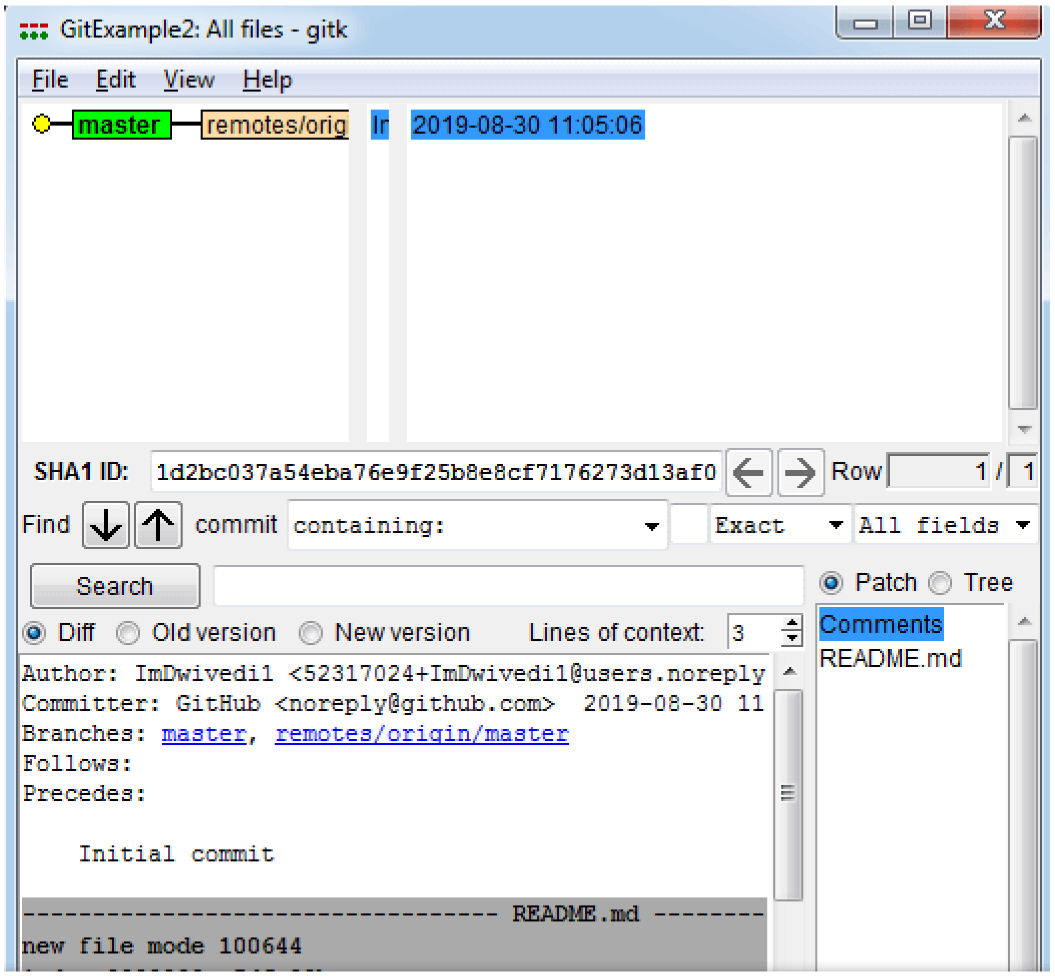


Gitk

gitk is like a special window that shows the history of your project in a graphical way. It's a user-friendly tool that helps you see what happened in the past and find specific things you're looking for in your project's history



```
rabbil@ICONS-01 MINGW64 ~
$ gitk

rabbil@ICONS-01 MINGW64 ~
$
```



```
2019-08-30 11:05:06
SHA1 ID: 1d2bc037a54eba76e9f25b8e8cf7176273d13af0   Row 1 / 1
Find commit containing:   Exact   All fields
Diff  Old version  New version   Lines of context: 3
Author: ImDwivedi1 <52317024+ImDwivedi1@users.noreply
Committer: GitHub <noreply@github.com>  2019-08-30 11
Branches: master, remotes/origin/master
Follows:
Precedes:

    Initial commit

---------------------------------- README.md --------
new file mode 100644
```

Terminology

1. **Repository (Repo)**: A repository is a directory or storage space where your project files are kept, along with the version history of those files.
2. **Commit**: A commit is a snapshot of your repository at a specific point in time. It represents the changes made to the files in your project.
3. **Branch**: A branch is a separate line of development within a repository. It allows you to work on features or fixes without affecting the main project until you're ready to merge your changes.
4. **Merge**: Merging is the process of combining the changes from one branch into another. It's typically used to integrate a feature branch back into the main branch.
5. **Pull**: Pulling is the process of fetching changes from a remote repository and integrating them into your local branch.
6. **Push**: Pushing is the process of sending your local commits to a remote repository, making them available to others.
7. **Remote**: A remote is a version of your repository that is hosted on a server, typically on a platform like GitHub, GitLab, or Bitbucket.
8. **Clone**: Cloning is the process of creating a copy of a remote repository on your local machine.
9. **Fork**: Forking is a process where you create a copy of a repository in your own GitHub account. It allows you to freely experiment with changes without affecting the original repository.
10. **Pull Request (PR)**: A pull request is a request to merge changes from one branch into another. It's typically used in open-source projects to propose changes and discuss them before merging.