



GIT VERSION CONTROL



ຊີວິດຈະງາຍຂຶ້ນຖ້າເຮົານຳໃຊ້ git version control





ສະແກນເພື່ອເອົາ ບິດຮຽນ



ຫົວຂໍ້ທີ່ຈະນຳສະເໜີ

01

Git ແນວທາງ

03

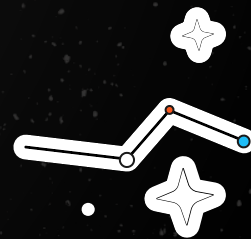
Github ແນວທາງ

02

ເປັນຫຍັງຕອງໃຊ້ Git

04

ວິທີການນຳໃຊ້ Git



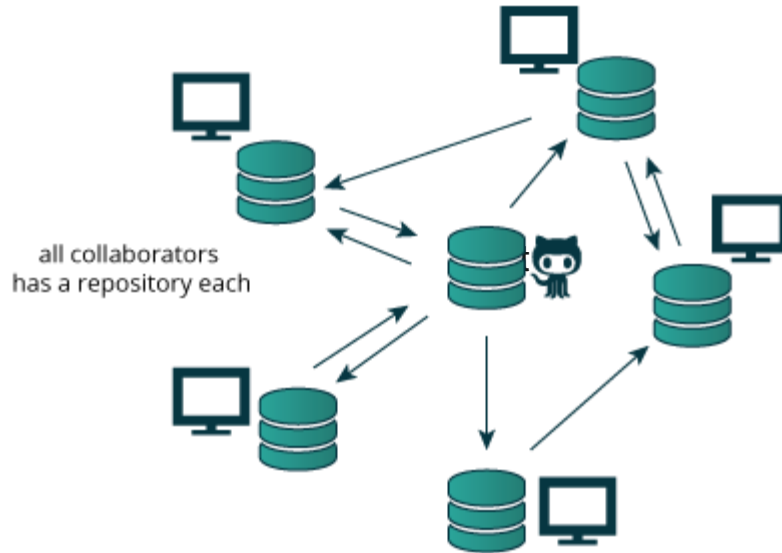
01

Git ແນ່ນຫຍັງ?

Git ເປັນ version control ຮູບແບບ
Distributed Version Control
System (DVCS) ໃຊ້ສໍາຫລັບຕິດຕາມການ
ກວດສອບແກໄຂ file ຫລື Source Code ເຊິ່ງ
ສາມາດຕວດສອບໄດ້ທຸກໂຕອັກສອນ ທຸກແຖວ ທຸກ
file ເຊິ່ງເຮົາສາມາດຮູ້ວ່າ file ຖືກແກໄຂເວລາໃດ
ແລະ ຜູ້ໃດເປັນຄົນແກ.



Distributed Version Control System (DVCS)





02

ເປັນຫຍັງຕອງໃຊ້ Git

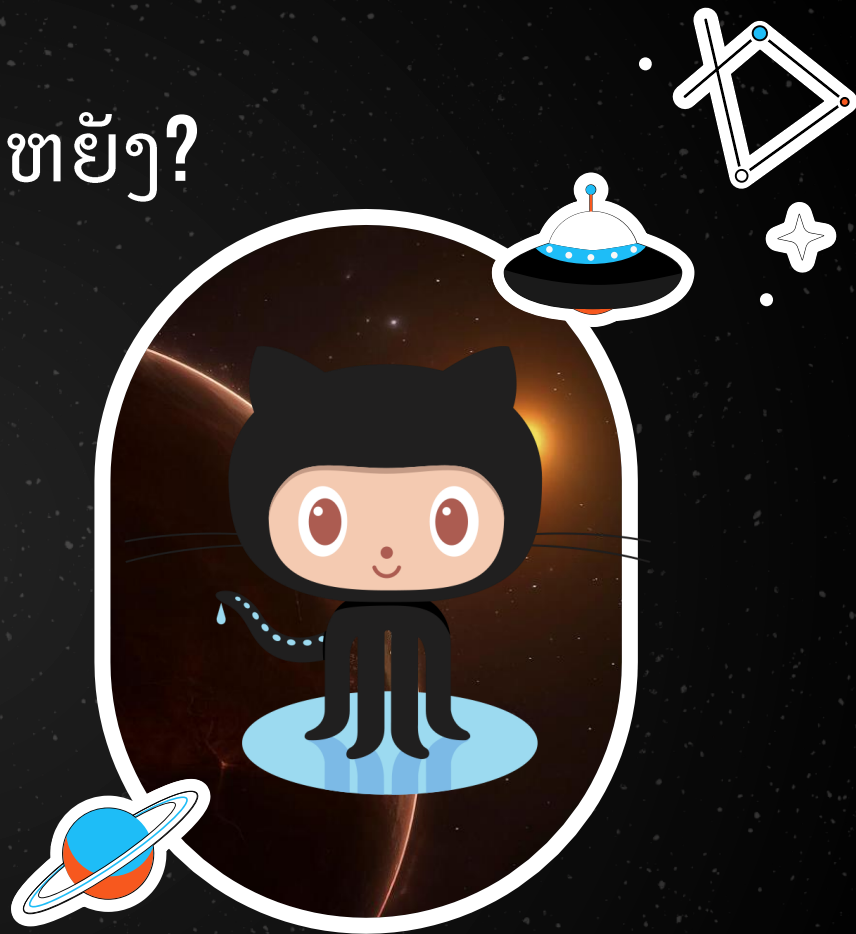
- Backup
- ຄັດລອກ file ເກົ່າ ແລະ ສາງ fileໃໝ່
- ເມື່ອຍ
- ຢາກຫາເຄື່ອງມືມາຊ່ວຍໃນການເກັບການປຸງແປງ
- ສາມາດຕິດຕາມການປຸງແປງຂອງ Code

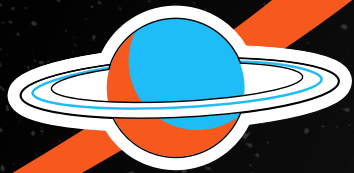


GITHUB ແມ່ນຫຍັງ?

Github ແມ່ນເວັບເຊີບເວີ ທີ່ໃຫ້ບໍລິການ
ໃນການຝາກfile Git ຫຼື Git ທີ່ເຮັດ
ວຽກເທິງເວັບໄຊ

ເຮັດໃຫ້ສາມາດໃຊ້ Git ຮວມກັບຄົນອື່ນ
ໂດຍຜ່ານເວັບໄຊ ເຊິ່ງມັກນິຍົມໃຊ້ເກັບ
Project Open Source

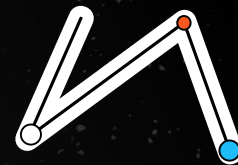




ວິທີການນຳໃຊ້ GIT



ຕິດຕັ້ງ GITBASH



Downloads



macOS



Windows



Linux/Unix

Older releases are available and the Git source repository is on GitHub.

Latest source Release

2.37.1

[Release Notes](#) (2022-07-04)

Download for Windows



ເລືອກລະບົບປະຕິບັດການ

Logos in PNG (bitmap) and EPS
formats are available for use in online
projects.





ສາງບັນຊີ Github ຫຼື Gitlap



Already have an account? [Sign in](#) →

Welcome to GitHub!
Let's begin the adventure

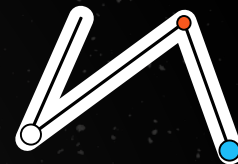
Enter your email



Continue

By creating an account, you agree to the [Terms of Service](#). For more information about GitHub's privacy practices, see the [GitHub Privacy Statement](#). We'll occasionally send you account-related emails.





CONFIG GIT USER

1. ເປີດ Terminal ຂຶ້ນມາ
- 2 ກວດສອບ git ວາຕິດຕັ້ງຫຼືຍັງ ໃຊ້ຄໍາສັ່ງ `Git -version`
- 3 `git config --global user.email "youremail@email.com"`
- 4 `git config --global user.name "username"`

```
C:\Users\KONG>git --version  
git version 2.27.0.windows.1
```

```
C:\Users\KONG>git config --global user.email "konglaodev@gamil.com"
```

```
C:\Users\KONG>git config --global user.name "konglaodev"
```

```
C:\Users\KONG>git config --global color.ui auto
```

```
C:\Users\KONG>
```



ເລີ່ມຕົ້ນການທົດລອງ



1 ເລີ່ມຕົ້ນຕໍາແໜ່ງ folder ຈາກ path ທີ່ຈະເລີ່ມຕົ້ນການເຮັດວຽກຂອງGit

- ເປີດ Terminal ຂຶ້ນມາ

- ສາງ folder ດວຍຄໍາສັ່ງ `mkdir ຊື່folder`

- ເຂົ້າໄປຕໍາແໜ່ງ folder `cd ຊື່folder`

- ເປີດ visual code ຕາມ path ປະຈຸບັນ ພິມ `Code` . Enter

- ສາງ file ຕາງໆລົງໃນ folder

- ເລີ່ມຕົ້ນ ນໍາໃຊ້ ຄໍາສັ່ງ `Git init`



ເລີ່ມຕົ້ນການທົດລອງ



C:\Windows\System32\cmd.exe

```
C:\Users\KONG\Desktop\git-workshop>mkdir yourfoldername
```

```
C:\Users\KONG\Desktop\git-workshop>dir
```

Volume in drive C has no label.

Volume Serial Number is DCE6-912A

Directory of C:\Users\KONG\Desktop\git-workshop

07/27/2022	06:50 AM	<DIR>	.
07/27/2022	06:50 AM	<DIR>	..
07/27/2022	06:41 AM	<DIR>	git_workshop
07/22/2022	07:03 PM	<DIR>	mysql-nodejs-crud-restful-api
07/23/2022	02:38 PM	<DIR>	node_crud
07/27/2022	06:50 AM	<DIR>	yourfoldername
		0 File(s)	0 bytes
		6 Dir(s)	39,853,559,808 bytes free

```
C:\Users\KONG\Desktop\git-workshop>cd yourfoldername
```

```
C:\Users\KONG\Desktop\git-workshop\yourfoldername>code .
```



ເລີ່ມຕົ້ນການທົດລອງ



✓ GIT_WORKSHOP

index.html U
main.css U
script.js U

TERMINAL

PROBLEMS

OUTPUT

DEBUG CONSOLE

COMMENTS

```
KONG@DESKTOP-TFLEASJ MINGW64 ~/Desktop/git-workshop/git_workshop
$ touch index.html
```

```
KONG@DESKTOP-TFLEASJ MINGW64 ~/Desktop/git-workshop/git_workshop
$ touch main.css
```

```
KONG@DESKTOP-TFLEASJ MINGW64 ~/Desktop/git-workshop/git_workshop
$ touch script.js
```

```
KONG@DESKTOP-TFLEASJ MINGW64 ~/Desktop/git-workshop/git_workshop
$ git init
Initialized empty Git repository in C:/Users/KONG/Desktop/git-workshop/git_workshop/.git/
```

```
KONG@DESKTOP-TFLEASJ MINGW64 ~/Desktop/git-workshop/git_workshop (master)
$ █
```



ຄໍາສັ່ງຕາງໆ



ສາງ Local repository

Git init

ເພີ່ມ file ໃນການເຮັດວຽກຂອງ Git

Git add filename

Git add *.css ເພີ່ມຫຼາຍຟາຍລ໌ພ້ອມນາມສະກຸນ

Git add . ເພີ່ມທັງໝົດ

ລຶບ file ຫຼື folder ອອກຈາກ Git

git rm -r --cached . ລຶບທັງໝົດ

git rm --cached <file_name> ລຶບຟາຍລ໌ດຽວ

ກວດສອບສະຖານະການປຸງແປງຂອງ repository
ພາຍໃນເຄື່ອງ

Git status



ຄໍາສັ່ງຕາງໆ



ເກັບປະຫວັດການແກໄຂຖາວອນ

Git Commit -m "message"

ເບິ່ງປະຫວັດການ Commit

Git log

Git log --oneline ສະແດງແຕ່ລະ log ແຖວດຽວ

Git log --graph ສະແດງເປັນເສັ້ນ Branch ໃຫ້ເບິ່ງງ່າຍ

ປຽບທຽບ version (Diff)

Git diff commitID ປຽບທຽບຕາມໄອດີ commit

Git diff commitid1 commitid2 ປຽບທຽບ2commit

ຍົກເລີກການແກໄຂ file

Git checkout filename

ການຍອນກັບ Version ໃຫ້ກັບໄປໃນສະຖາກອນທີ່ຈະ add file

ເຂົ້າມາ repository

Git reset

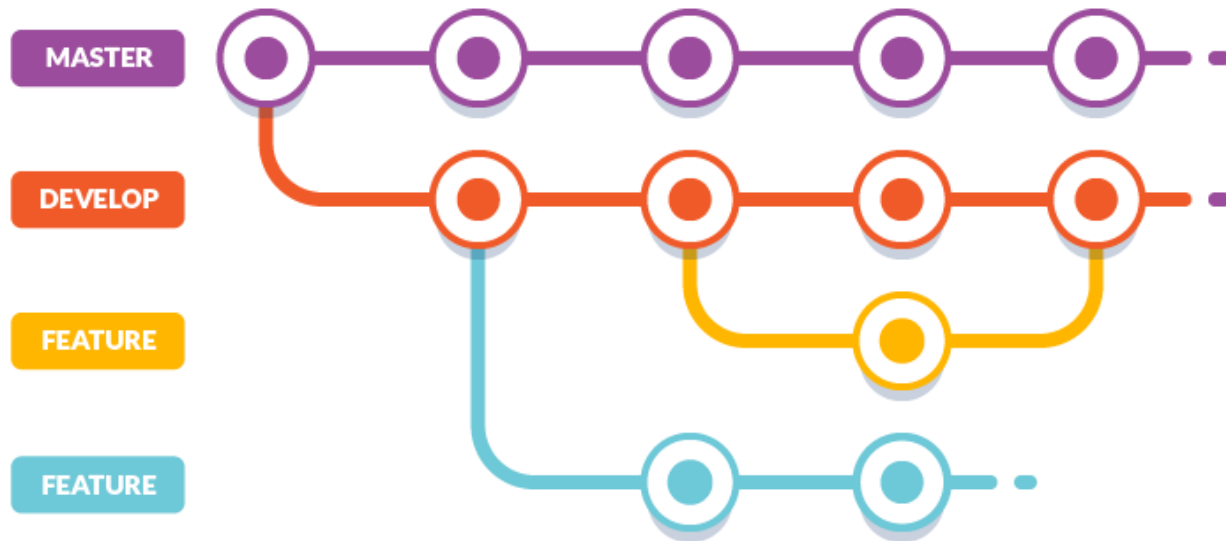
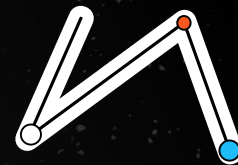
ຄໍາສັ່ງໃນການຍອນກັບຄືນເວີຊັນ

Git reset --option commitid

--option (soft, Mixed, hard)



Git Branch



Git Branch



ການສະແດງຊື່ Branch

Git branch

ການສາງ Branch

Git checkout -b name

ການລຶບ Branch

Git branch -d nameTodelete

ການສະຫຼັບໄປ Branch ໃໝ່

Git checkout nameBranch

ການລວມ Branch

Git merge nameBranchTomerge



Git Remote Repository

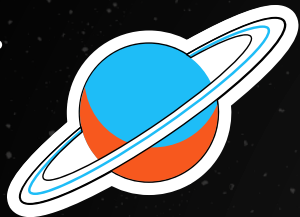


1 **Git push** ເປັນຄໍາສັ່ງທີ່ໃຊ້ໃນການນຳສິ່ງທີ່ຢູ່ໃນເຄື່ອງຂອງເຮົາ (Local Repository) ໄປອັບເດດໃຫ້ກັບ Remote Repository (Server)

2 **Git pull** ເປັນຄໍາສັ່ງທີ່ໃຊ້ດຶງຂໍ້ມູນຈາກ Repository ຈາກ server ມາເກັບຫຼືອັບເດດການແກ້ໄຂລາສຸດມາທີ່ເຄື່ອງຂອງເຮົາ

3 **Git clone** ເປັນການນຳເອົາໂປເຈັກທີ່ເກັບໄວ້ທີ່ repository ເທິງ Server ມາເກັບໄວ້ໃນເຄື່ອງຂອງເຮົາ ຫຼື ສະມາຊິກໃນທີມ





ຂອບຈາຍຍ!

