



TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Nhập môn Công nghệ phần mềm Introduction to Software Engineering (IT3180)

Bài tập tuần 05 Quản lý cấu hình phần mềm & Thực hành với công cụ GIT

Thông tin GV

Mục tiêu

- Thực hiện các bài tập (câu hỏi) về nội dung Quản lý cấu hình phần mềm
- Thực hành với công cụ GIT

Đánh giá

- Hoàn thành các bài tập về nội dung Quản lý cấu hình phần mềm, nắm được khái niệm, các hoạt động và công cụ quản lý cấu hình
- Hoàn thành bài thực hành với công cụ GIT

Bài 1.1

a) Loại nào sau đây là một phần của đầu ra của quy trình phần mềm?

- 1. Chương trình máy tính**
- 2. Tài liệu mô tả các chương trình máy tính**
- 3. Dữ liệu**
- 4. Tất cả những phương án trên**

Bài 1.1

b) Khái niệm quản lý cấu hình phần mềm nào giúp chúng ta kiểm soát sự thay đổi mà không cản trở đến những thay đổi chính đáng?

- 1. Baselines**
- 2. Source code**
- 3. Data model**
- 4. Các phương án trên đều sai**

Bài 1.1

c) Quá trình nào sau đây tập hợp các thành phần chương trình, dữ liệu và thư viện, sau đó biên dịch và liên kết các thành phần này để tạo ra một hệ thống thực thi?

- 1. Version management**
- 2. Release management**
- 3. Change management**
- 4. System building**

d) Hoạt động nào sau đây không phải là Hoạt động quản lý cấu hình phần mềm?

- 1. Configuration item identification**
- 2. Risk management**
- 3. Release management**
- 4. Branch management**

e) Những gì liên quan đến việc chuẩn bị phần mềm để phát hành bên ngoài và theo dõi các phiên bản hệ thống đã được phát hành cho khách hàng sử dụng?

- 1. System building**
- 2. Release management**
- 3. Change management**
- 4. Version management**

Bài 1.2

- a) Mục đích chính của Quản lý cấu hình phần mềm (SCM) là gì?
- b) Tên của quá trình sẽ đảm bảo rằng các phiên bản khác nhau của hệ thống và các thành phần của nó được ghi lại và duy trì?
- c) Những công việc chính của Quản lý cấu hình phần mềm (SCM) là gì?
- d) Một mô hình kho lưu trữ (repository) được sử dụng để làm gì? Các yêu cầu về kho lưu trữ (repository) có giống những yêu cầu đối với cơ sở dữ liệu điển hình không?
- e) Hãy lấy ví dụ về các đối tượng cấu hình cơ bản do kỹ sư phần mềm tạo ra trong một số giai đoạn của quá trình phát triển phần mềm?



TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

Công nghệ: GIT

Sơ lược về Git

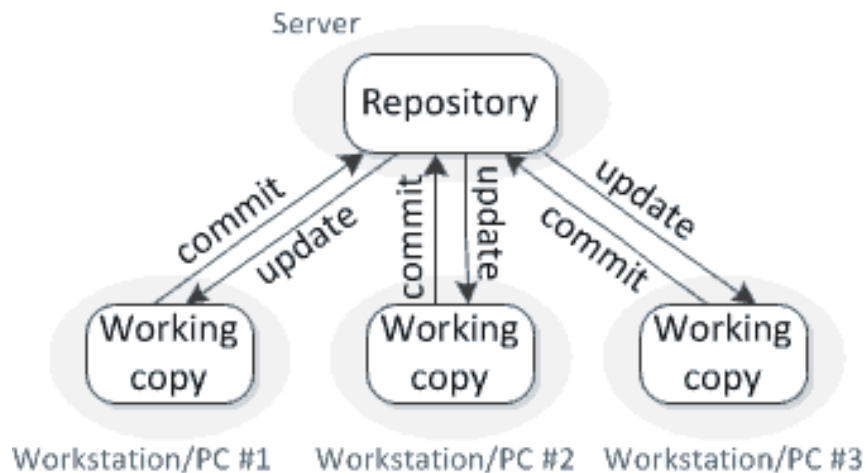
- Khái niệm: là một hệ thống quản lí phiên bản phân tán
- Là phần mềm quản lí mã nguồn phổ biến nhất
- Dễ sử dụng, an toàn và nhanh chóng



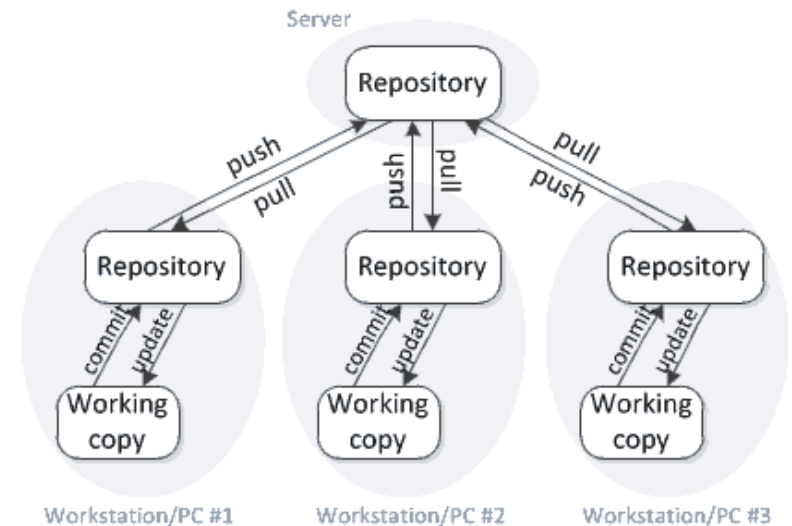
Sơ lược về Git

- Quản lý phiên bản (version control) tập trung và phân tán

Centralized version control



Distributed version control





Required for any path

Git - Version Control

Basic Terminal Usage

Data Structures & Algorithms

GitHub

Licenses

Semantic Versioning

SSH

HTTP/HTTPS and APIs

Design Patterns

Character Encodings

Web Developer in 2020

Choose your path

Front-end

Back-end

DevOps

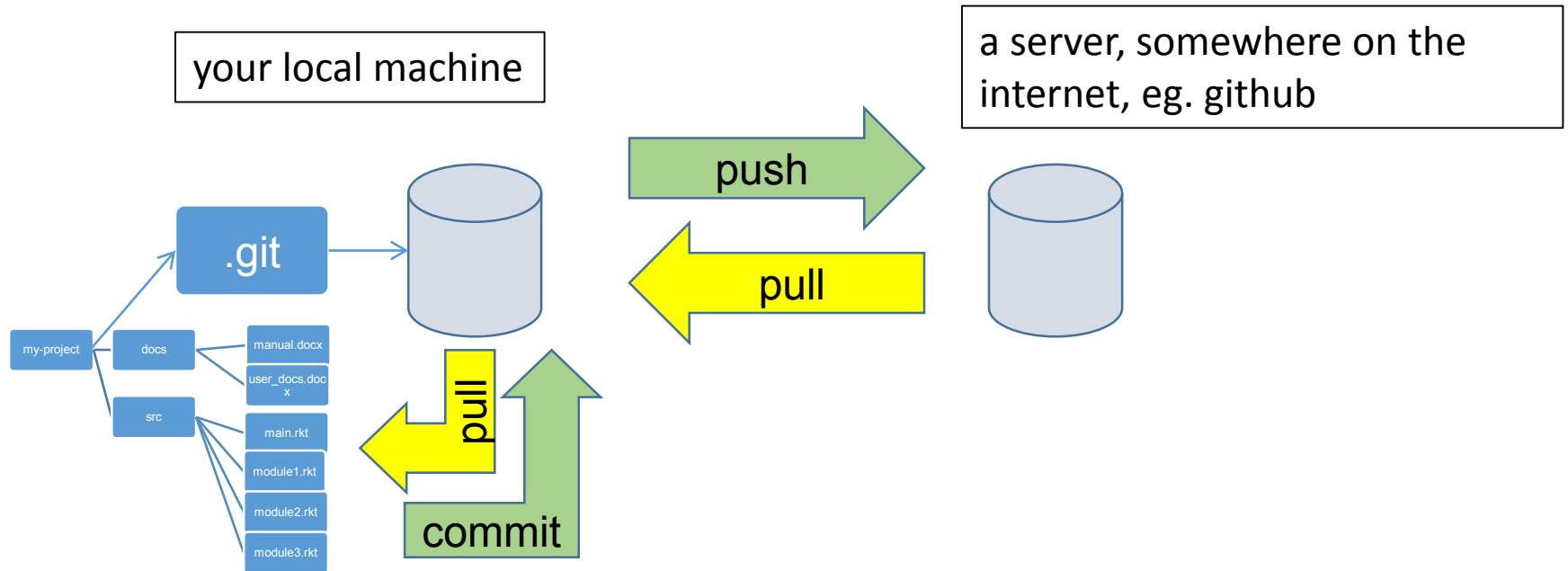
Find the detailed version of this roadmap along with resources and other roadmaps

<http://roadmap.sh>

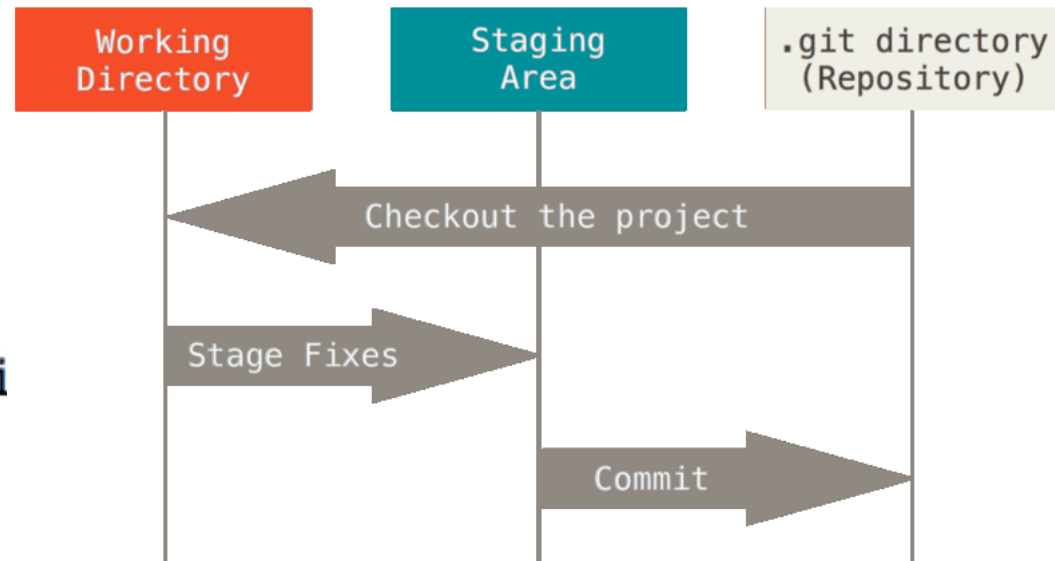
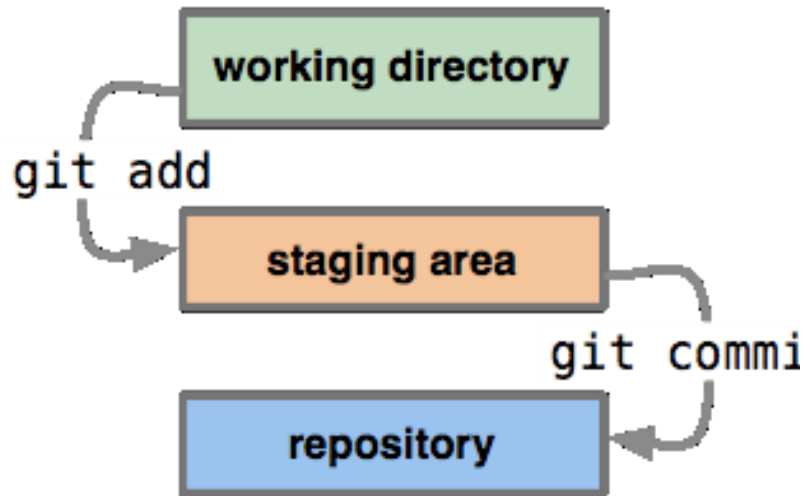
Quy trình sản xuất phần mềm với Git

1. Cài đặt Git
2. Khởi tạo git trong project
3. Làm việc với branch
4. Xử lý merge request
5. Vấn đề conflict
6. Deploy server dùng git *

The whole picture



Working Directory, Staging, Committed



Working Directory, Staging, Committed

- Các thuật ngữ:

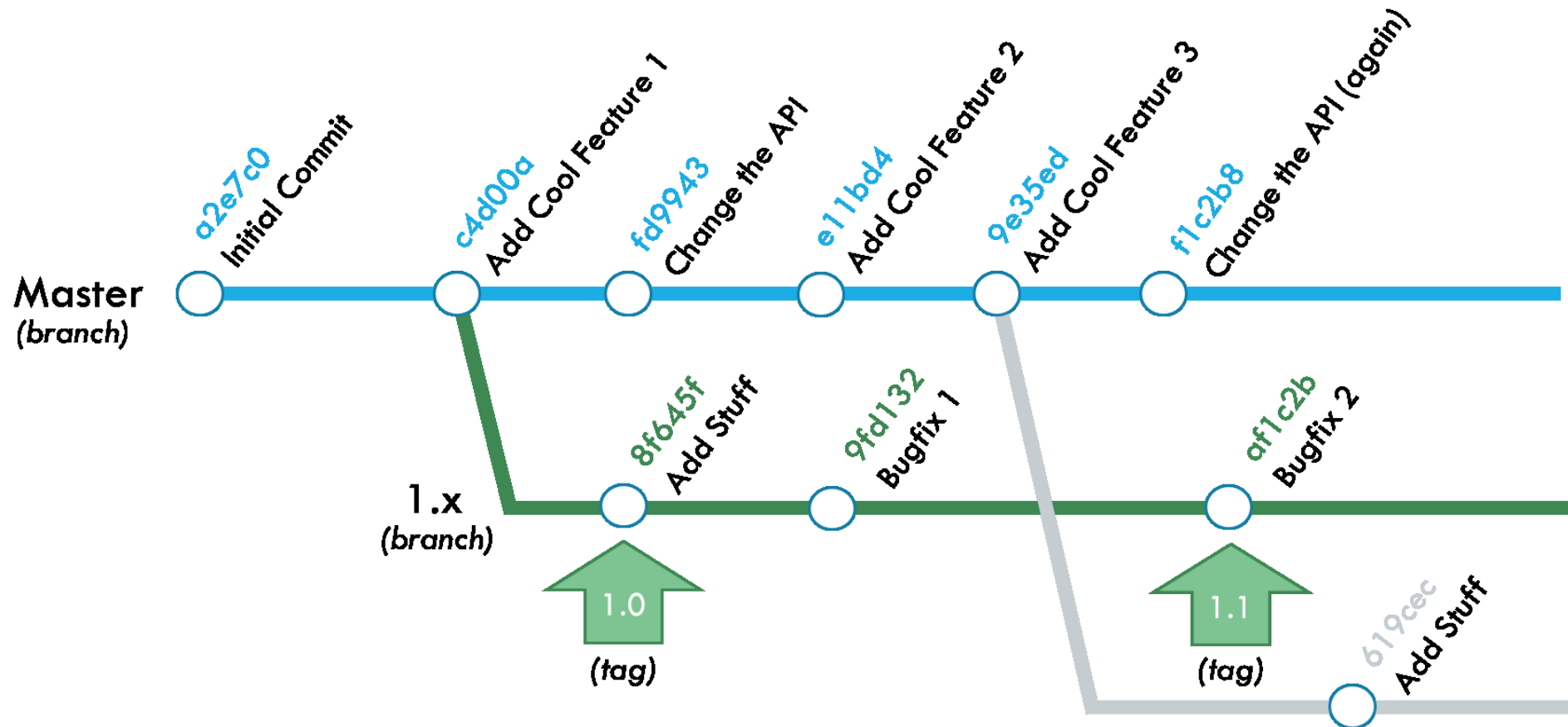
- Branch

- Lịch sử của những thay đổi liên tiếp đối với mã nguồn
- Một nhánh mới có thể được tạo bất kỳ lúc nào, từ bất kỳ commit hiện có
- Có thể đại diện cho các phiên bản của mã
 - Version 1.x, 2.x, 3.x, etc.
- Có thể đại diện cho các bản sửa lỗi nhỏ / phát triển tính năng
- Branches are cheap
 - Chuyển đổi nhanh chóng
 - Dễ dàng "merge" vào các chi nhánh khác
 - Dễ tạo, dễ hủy

- Tag

- Biểu diễn một single commit
- Thường thân thiện với người dùng
 - Version Numbers
- Một Repository có thể được tạo bằng cách:
 - Nhân bản một cái hiện có ([git clone](#))
 - Tạo một cái cục bộ (locally) mới ([git init](#))

Working Directory, Staging, Committed



Working Directory, Staging, Committed

- Các thuật ngữ:
- Commit
 - Một snapshot cụ thể trong cây phát triển
 - Tập hợp các thay đổi được áp dụng cho các tệp của dự án
 - Text changes, File and Directory addition/removal, chmod
 - Metadata về thay đổi
 - Được xác định bởi **SHA-1 Hash**
 - Có thể được rút ngắn thành khoảng. 6 ký tự để sử dụng CLI
 - (e.g., “`git show 5b16a5`”)
 - **HEAD** –commit gần đây nhất
 - **ORIG_HEAD** – sau khi merge, **HEAD** trước đó
 - **<commit>~n** – commit thứ **n**th trước **<commit>**
 - e.g., `5b16a5~2` or `HEAD~5`
 - **master@{01-Jan-2018}** – commit cuối cùng trên nhánh **master** branch trước January 1, 2018

Working Directory, Staging, Committed

- Các thuật ngữ:

- **Working Files**

- Các tệp tin hiện thời trong File System

- **The Stage (còn gọi là “index”)**

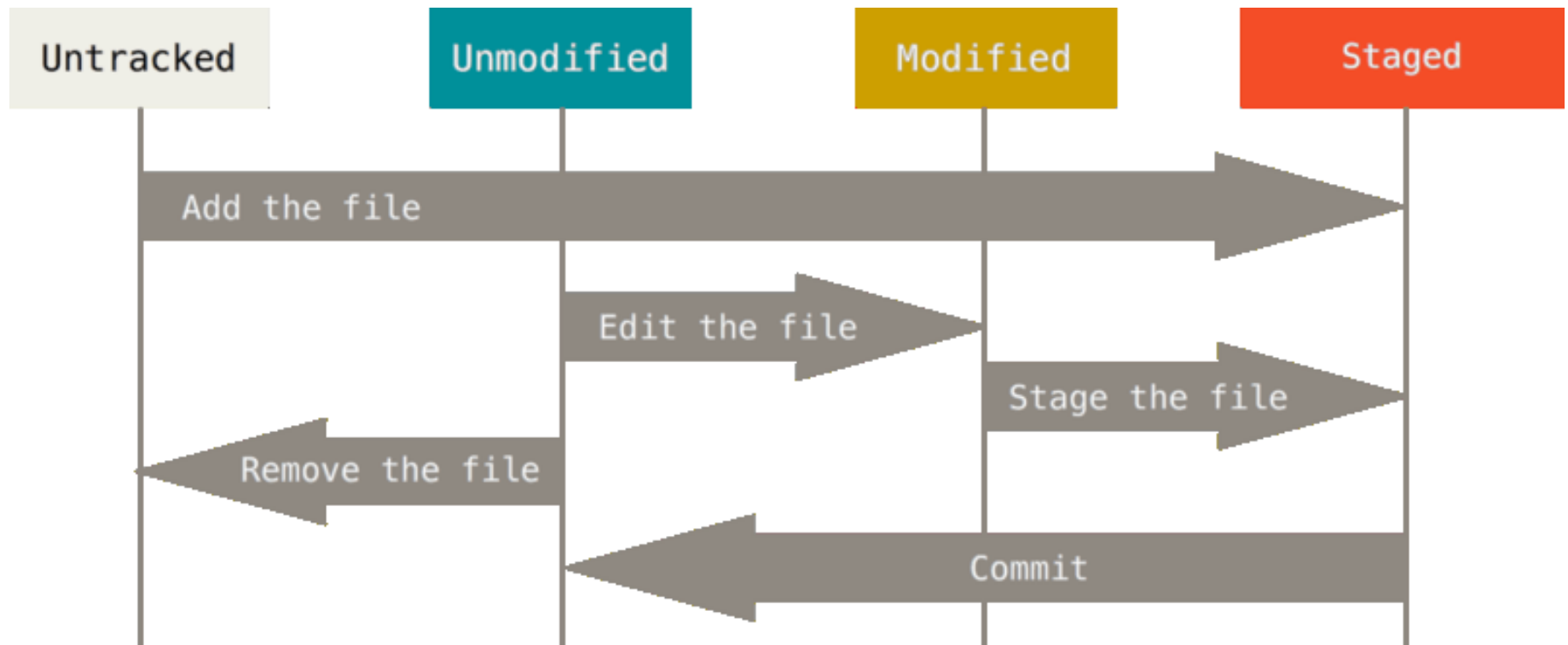
- Staging là bước đầu tiên để tạo một commit
- stage được dùng để báo cho Git biết những thay đổi nào được bao gồm trong commit
- Các thay đổi tệp phải được “**added**” vào stage một cách tường minh
- Chỉ những thay đổi được **staged** thì sẽ được **committed**

- **Checkout**

- Thay thế các tệp đang làm việc hiện tại bằng các tệp từ một **branch** cụ thể hoặc **commit**

- Sử dụng “**git diff**” để xem những thay đổi nào tồn tại.
- Sử dụng “**git add**” để nói với Git rằng các thay đổi của tệp phải được thêm vào **stage**.
- Sử dụng “**git status**” để xem các thay đổi trong tệp làm việc của bạn, cũng như những thay đổi nào đã được thực hiện cho commit.
- Sử dụng “**git commit**” để chuyển đổi tất cả các thay đổi theo giai đoạn thành một commit.
 - **git commit -m “my commit message”**
 - **git commit -m “my commit message” -a**
 - Sẽ tự động sắp xếp tất cả các tệp được repo theo dõi và thêm chúng vào commit.
 - **Không nên làm điều này.**

Working Directory, Staging, Committed

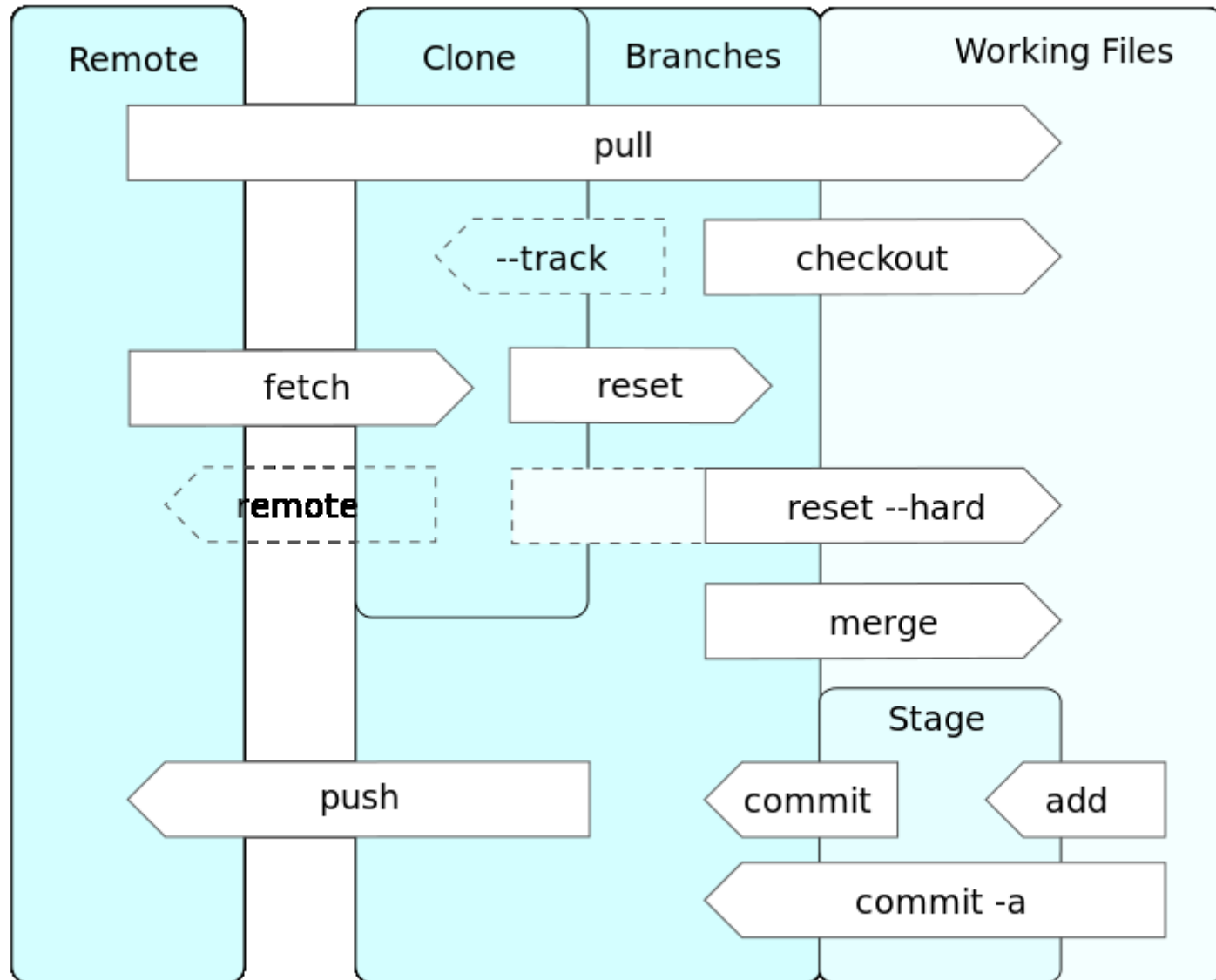


Workflow tổng quát

1. Chọn một bug/feature để giải quyết
 2. Checkout nhánh phù hợp
 3. Pull các bổ sung mới nhất từ kho lưu trữ chính
 4. Tạo một branch cho bug/feature yêu cầu
 5. Thực hiện viết code / lập trình & commit
 6. Push các thay đổi đến các kho lưu trữ từ xa (remote repo.)
 7. Tạo một pull request đến main repo
- pull request là nơi các lập trình viên khác xem xét mã của bạn và đưa ra nhận xét / đề xuất.
 - Nếu cần thay đổi mã của bạn, hãy lặp lại quy trình.

Không commit trực tiếp vào main repo.

Working Directory, Staging, Committed



1. Cài đặt Git

1. Download git: <https://git-scm.com/downloads>
2. Kiểm tra version

```
git --version
```

3. Cấu hình thông tin cá nhân

```
git config --global user.name "Name"  
git config --global user.email "Email"
```

4. SSH keys

2. Khởi tạo git trong project

- Khởi tạo

```
git init // khởi tạo thư mục .init  
git clone [url] [folder_name] // clone source code từ url
```

- Kiểm tra repository

```
git remote -v // xem các liên kết  
git remote set-url origin [url] // thay đổi liên kết
```

- .gitignore

Một số lệnh cơ bản

<code>git add [-A . -u]</code>	<code>// thêm / thay đổi file, thư mục</code>
<code>git status</code>	<code>// kiểm tra trạng thái của project</code>
<code>git commit -m "Content commit"</code>	<code>// ghi lại việc thêm / thay đổi</code>
<code>git push origin [branch]</code>	<code>// thêm commit vào kho lưu trữ</code>
<code>git pull</code>	<code>// cập nhật kho lưu trữ</code>

Một số lệnh quan trọng

git merge/ rebase	// ghép 2 nhánh
git reset	// xóa những cập nhật hiện tại
git checkout	// chuyển nhánh
git stash	// lưu lại thay đổi chưa commit
git log	// xem lịch sử commit
git revert	// quay lại trạng thái trước đó

```
>> git clone git@github.com:user/project.git
```

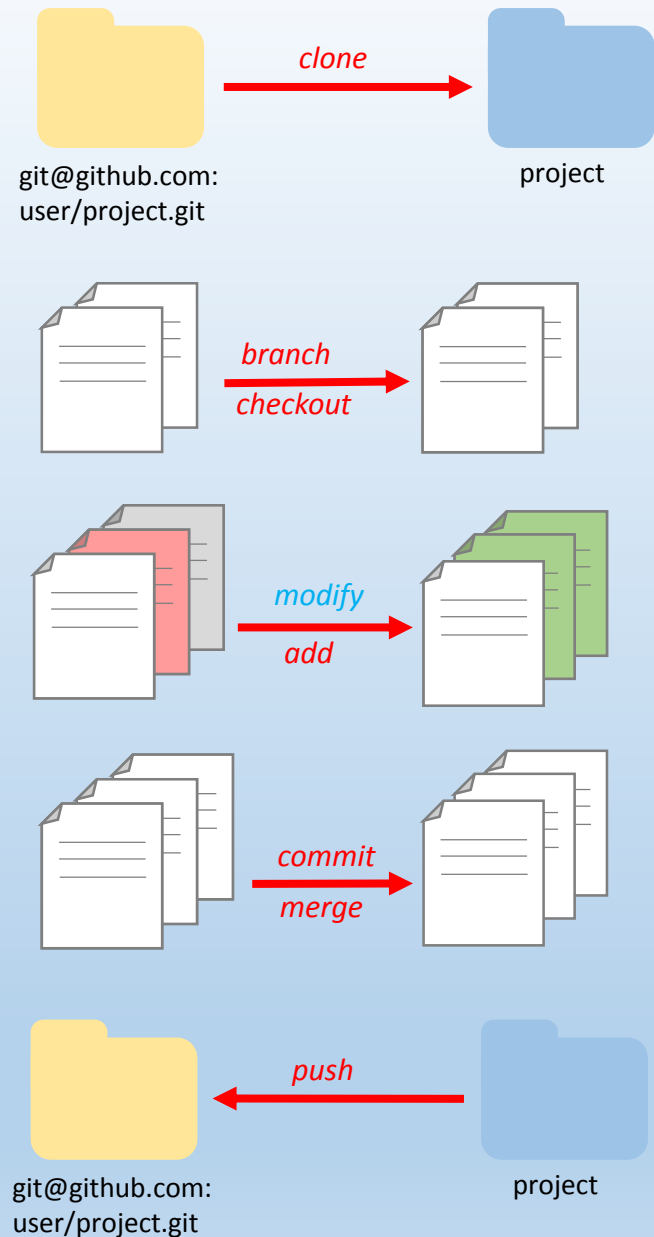
```
>> cd project  
>> ls  
file1  
file2
```

```
>> git checkout -b bugfixBranch  
>> touch file3  
>> echo "new code" >> file1  
>> git status  
modified:   file1  
untracked files: file3  
>> git add .  
>> git status  
modified: file1  
new file: file3  
>> git commit -m "fixed stuff"
```

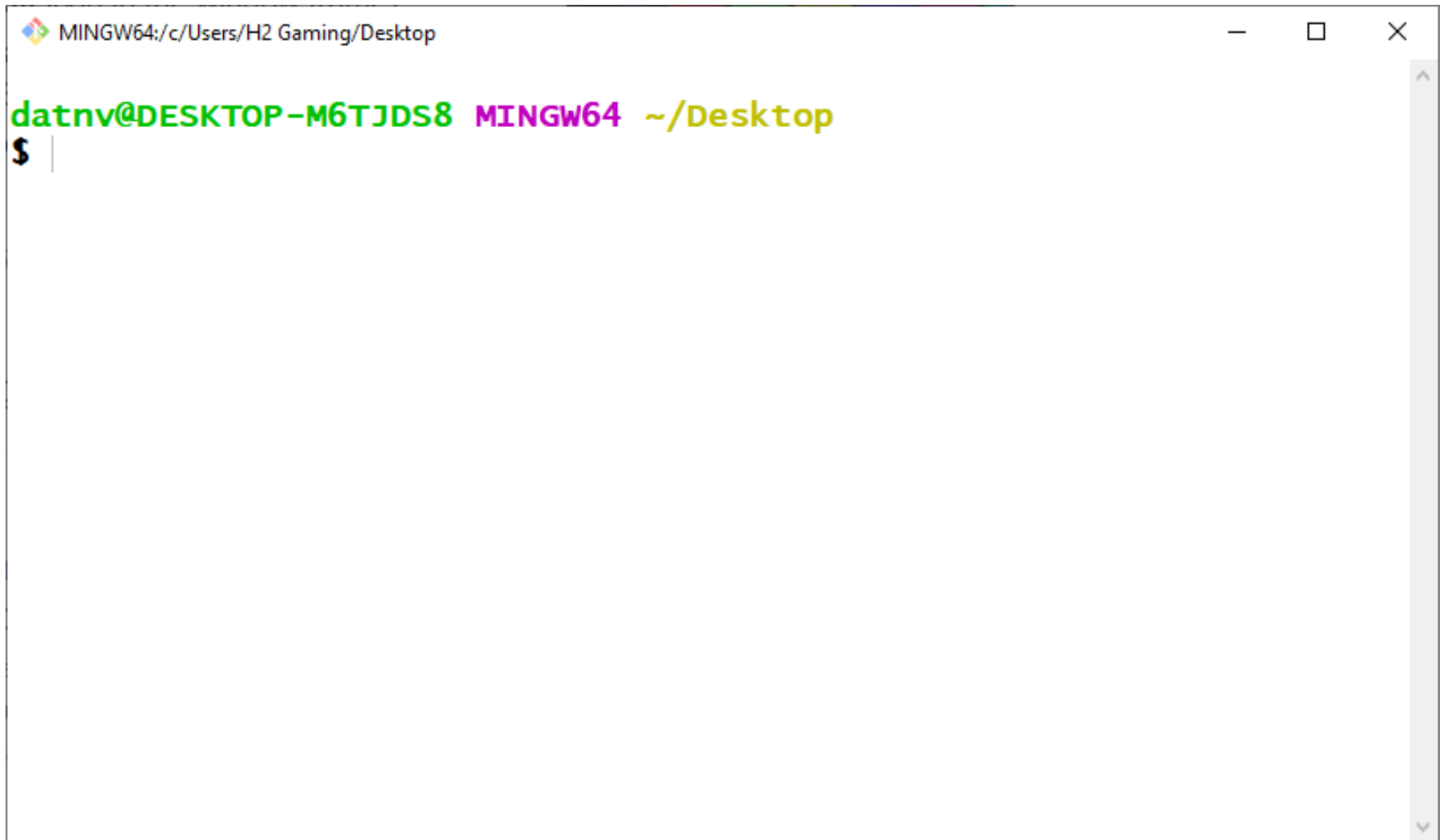
```
>> git checkout master  
>> git diff bugfixBranch  
modified file1  
+fixed stuff  
created file3  
>> git merge bugfixBranch
```

```
>> git push origin master
```

```
>> git request-pull bigProjectv1.0 \ https://github.com/user/project  
master \  
> pullrequest.txt
```



Các câu lệnh cơ bản



```
MINGW64:/c/Users/H2 Gaming/Desktop
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop
$
```

Các câu lệnh cơ bản



```
MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop
$ mkdir Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop
$ cd Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git
$
```

Tạo thư mục Git

Di chuyển đến Git/

Các câu lệnh cơ bản



```
MINGW64:/c:/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git
$ git init
Initialized empty Git repository in C:/Users/H2 Gaming/Desktop/Git/.git/

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$
```

Khởi tạo .init

Các câu lệnh cơ bản

```
MINGW64; c:/Users/H2 Gaming/Desktop/Git
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git
$ git init
Initialized empty Git repository in C:/Users/H2 Gaming/Desktop/Git/.git/

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)

        README.md

nothing added to commit but untracked files present (use "git add"
to track)

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$
```

Kiểm tra trạng thái

Các câu lệnh cơ bản

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git add -A

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git commit -m "Init"
[master (root-commit) b06a567] Init
1 file changed, 1 insertion(+)
create mode 100644 README.md

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$
```

Cập nhật thay đổi

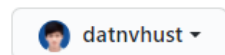
Tạo ra 1 commit

Các câu lệnh cơ bản

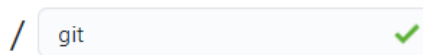
Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner *



Repository name *



Great repository names are short and memorable. Need inspiration? How about [vigilant-parakeet](#)?

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☐ **Add a README file**

This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**

Choose which files not to track from a list of templates. [Learn more.](#)

☐ **Choose a license**

A license tells others what they can and can't do with your code. [Learn more.](#)

Create repository

Các câu lệnh cơ bản

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git add -A

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git commit -m "Init"
[master (root-commit) b06a567] Init
 1 file changed, 1 insertion(+)
 create mode 100644 README.md

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git remote add origin https://github.com/datnvhurst/git.git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ |
```

Kết nối với Github

Các câu lệnh cơ bản

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git
$ git commit -m "Init"
[master (root-commit) b06a567] Init
 1 file changed, 1 insertion(+)
 create mode 100644 README.md


datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git remote add origin https://github.com/datnvhurst/git.git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git push origin -u master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 221 bytes | 221.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/datnvhurst/git.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$
```


Đưa commit lên
kho lưu trữ

Các câu lệnh cơ bản

 datnvhus / git

<> Code

! Issues


 Pull requests


▶ Actions


 Projects


 Wiki

 Security

 Insights

 Settings

 master ▾

 1 branch

 0 tags


Go to file

Add file ▾

 Code ▾



Nguyễn Văn Đạt Init

b06a567 3 minutes ago  1 commits



README.md

Init

3 minutes ago

README.md



##Init

3. Làm việc với nhánh (Branch)

- Branch chính
 - Master
 - Developer
 - Staging
- Branch hỗ trợ
 - Tính năng
 - Release
 - Sửa lỗi (hotfix)

3. Làm việc với nhánh (Branch)

- Nhánh master: lưu trữ lịch sử phát hành chính thức (production)
- Nhánh develop: Nhánh tích hợp cho các tính năng (dev)
- Nhánh staging: được tách ra từ develop để QA/QC kiểm tra

Thao tác với nhánh cơ bản

Chuyển nhánh khác

```
Git checkout [branch]  
Git checkout -b [new_branch]
```

```
Git checkout [branch]  
Git checkout -b [new_branch]
```

4. Xử lí merge request

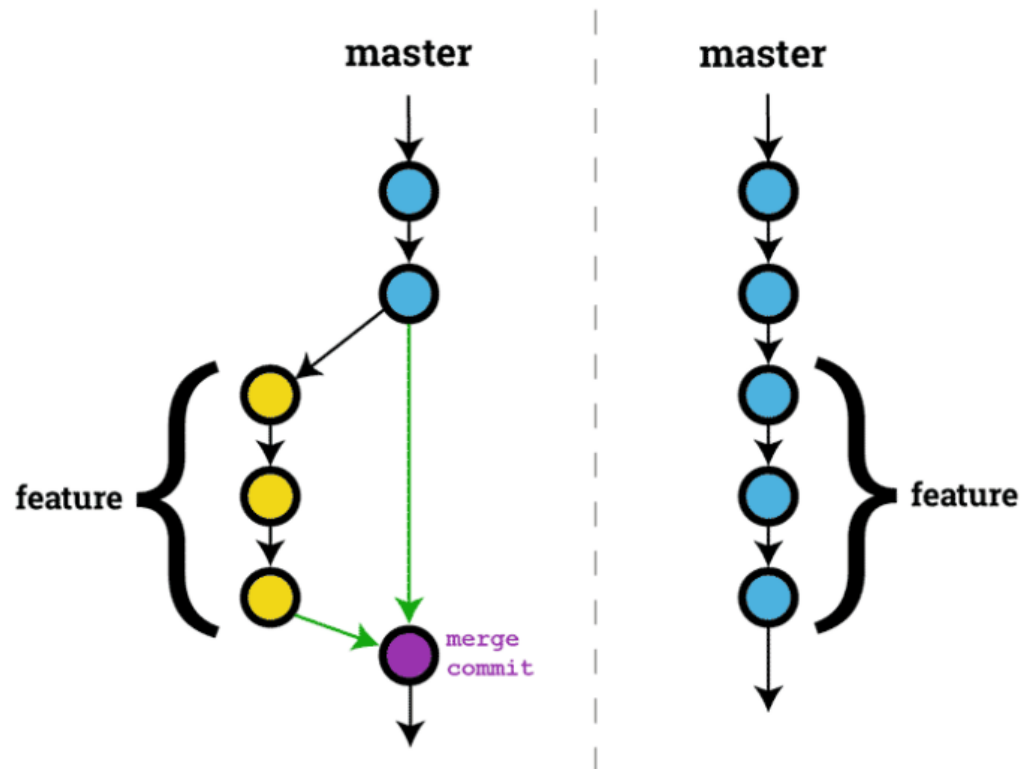
Merge

- Tích hợp 2 nhánh
- Tạo ra 1 merge commit và thêm vào nhánh đích
- Cây cực kì rối, khó nhìn
- Tìm kiếm và sửa lỗi dễ dàng

Rebase

- History dễ nhìn, thẳng hàng
- Đem từng commit ở nhánh nguồn đặt lên nhánh chính theo đúng thứ tự
- Dễ gây xung đột

Merge & Rebase



Merge & Rebase

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git remote add origin https://github.com/datnvhusit/git.git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git push origin -u master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 221 bytes | 221.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/datnvhusit/git.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git checkout -b develop
Switched to a new branch 'develop'

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$
```

Chuyển nhánh
develop

Merge & Rebase

MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)

\$ git status

On branch develop

Untracked files:

(use "git add <file>..." to include in what will be committed)

index.html

nothing added to commit but untracked files present (use "git add" to track)

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)

\$

Merge & Rebase

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git add -A

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git commit -m "Add index.html"
[develop 59b344a] Add index.html
1 file changed, 11 insertions(+)
create mode 100644 index.html

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git push origin develop
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 448 bytes | 448.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/datnvhust/git/pull/new/develop
remote:
To https://github.com/datnvhust/git.git
 * [new branch]      develop -> develop

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
```

Merge & Rebase

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ |
```

Chuyển nhánh
master

Merge & Rebase

```
MINGW64:/c:/Users/H2 Gaming/Desktop/Git
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git merge develop
Updating b06a567..59b344a
Fast-forward
 index.html | 11 ++++++++
 1 file changed, 11 insertions(+)
 create mode 100644 index.html

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ |
```

Merge nhánh
develop vào master

Merge & Rebase

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git merge develop
Updating b06a567..59b344a
Fast-forward
 index.html | 11 ++++++++
 1 file changed, 11 insertions(+)
 create mode 100644 index.html

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git push origin master
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/datnvhusit/git.git
 b06a567..59b344a  master -> master


datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$
```

Merge & Rebase

 datnvhus / git

<> Code


! Issues


 Pull requests


▶ Actions


 Projects


 Wiki

 Security

 Insights

 Settings

 master ▾

 2 branches

 0 tags

Go to file

Add file ▾

 Code ▾



Nguyễn Văn Đạt Add index.html

59b344a 2 minutes ago ⌚ 2 commits



README.md

Init

8 minutes ago



index.html

Add index.html

2 minutes ago

README.md



##Init

- Xảy ra khi hai hay nhiều người thay đổi cùng một tệp
- Phát hiện khi xử lý merge request

```
Accept Current Change | Accept Incoming Change | Accept Both Changes |
<<<<<< HEAD (Current Change)
    <div>Github</div>
=====
    <div>Gitlab</div>
>>>>>> develop (Incoming Change)
```

MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)

\$ git status

On branch master

Your branch is up to date with 'origin/master'.

Changes not staged for commit:

(use "git add <file>..." to update what will be committed)

(use "git checkout -- <file>..." to discard changes in working directory)

modified: index.html

no changes added to commit (use "git add" and/or "git commit -a")

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)

\$

MINGW64:/c/Users/H2 Gaming/Desktop/Git

— □ ×

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)

\$ git add -A

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)

\$ git commit -m "Change index in master"

[master 4a6cbe7] Change index in master

1 file changed, 1 insertion(+), 1 deletion(-)

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)

\$ git push origin master

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 346 bytes | 346.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/datnvhust/git.git

59b344a..4a6cbe7 master -> master

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)

\$

MINGW64:/c/Users/H2 Gaming/Desktop/Git

```
Writing objects: 100% (3/3), 346 bytes | 346.00 KiB/s, done.  
Total 3 (delta 1), reused 0 (delta 0)  
remote: Resolving deltas: 100% (1/1), completed with 1 local object.  
To https://github.com/datnvhust/git.git  
    59b344a..4a6cbe7  master -> master
```

```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
```

```
$ git checkout develop  
Switched to branch 'develop'
```

```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
```

```
$ git status  
On branch develop  
Changes not staged for commit:  
  (use "git add <file>..." to update what will be committed)  
  (use "git checkout -- <file>..." to discard changes in working directory)
```

```
        modified:   index.html
```

```
no changes added to commit (use "git add" and/or "git commit -a")
```

```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
```

```
$
```

MINGW64:/c/Users/H2 Gaming/Desktop/Git

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)

\$ git add -A

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)

\$ git commit -m "Change index.html in develop"

[develop c7cb391] Change index.html in develop
1 file changed, 1 insertion(+), 1 deletion(-)

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)

\$ git push origin develop

Enumerating objects: 5, done.

Counting objects: 100% (5/5), done.

Delta compression using up to 4 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 351 bytes | 351.00 KiB/s, done.

Total 3 (delta 1), reused 0 (delta 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/datnvhus/git.git

59b344a..c7cb391 develop -> develop

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)

\$

```
MINGW64:/c/Users/H2 Gaming/Desktop/Git
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 351 bytes | 351.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/datnvhust/git.git
   59b344a..c7cb391  develop -> develop

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (develop)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git merge develop
Auto-merging index.html
CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master|MERGING)
$
```

master|MERGING??

FileEditSelectionViewGoRunTerminalHelp

index.html - Git - Visual Studio Code

EXPLORERGIT

index.html

README.md

index.html

1<!DOCTYPE html>

2<html lang="en">

3<head>

4<meta charset="UTF-8">

5<meta name="viewport" content="width=device-width, initial-scale=1.0">

6<title>Document</title>

7</head>

8<body>

9<<<<<< HEAD (Current Change)

10<div>Github</div>

11=====

12<div>Gitlab</div>

13>>>>>> develop (Incoming Change)

14</body>

15</html>

master!

0 0

Live Share

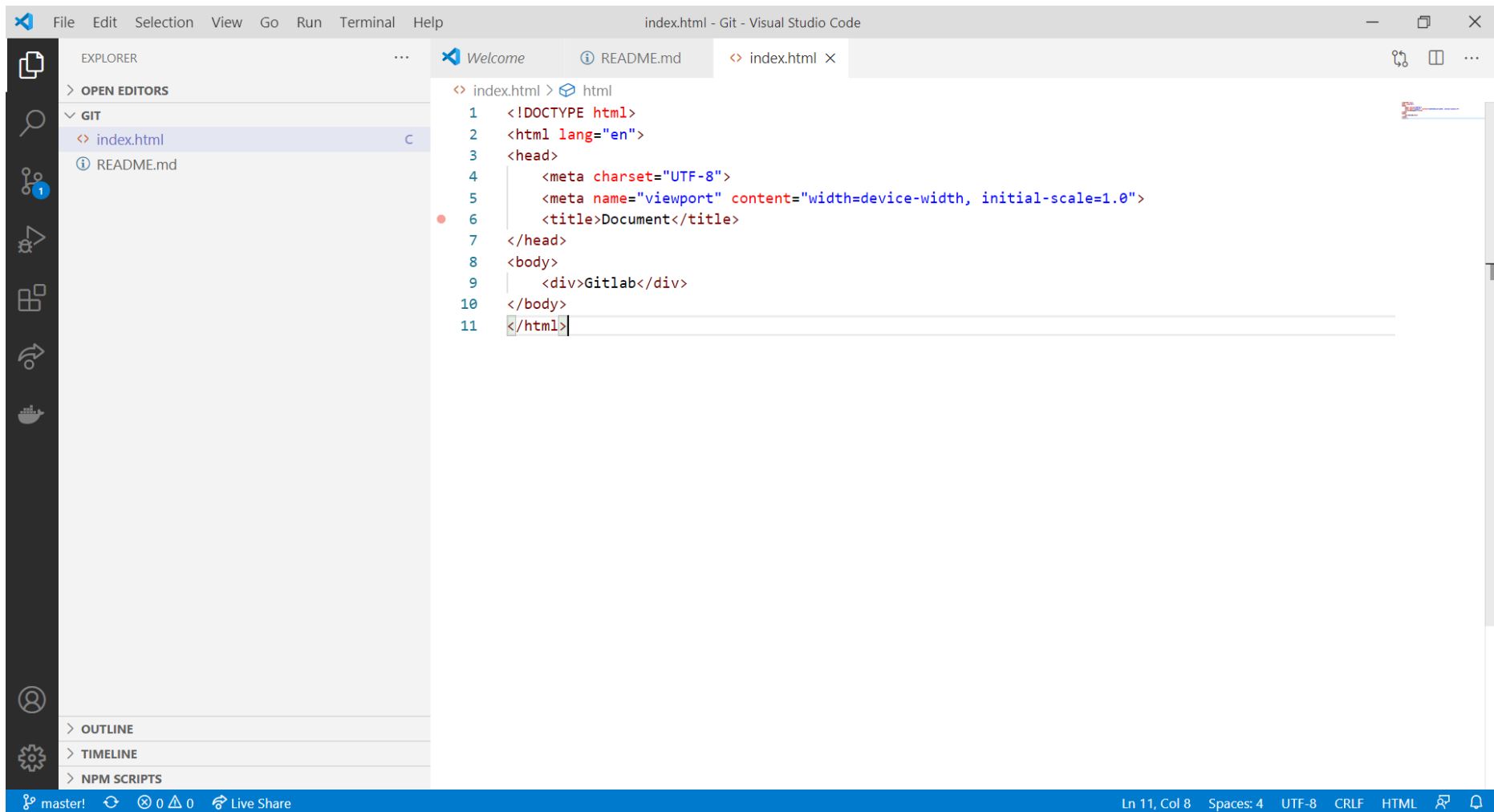
Ln 15, Col 8

Spaces: 4

UTF-8

CRLF

HTML



MINGW64:/c/Users/H2 Gaming/Desktop/Git

CONFLICT (content): Merge conflict in index.html
Automatic merge failed; fix conflicts and then commit the result.

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master|MERGING)

\$ git status

On branch master

Your branch is up to date with 'origin/master'.

You have unmerged paths.

(fix conflicts and run "git commit")

(use "git merge --abort" to abort the merge)

Unmerged paths:

(use "git add <file>..." to mark resolution)

both modified: index.html

no changes added to commit (use "git add" and/or "git commit -a")

datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master|MERGING)

\$

MINGW64:/c/Users/H2 Gaming/Desktop/Git

both modified: index.html

no changes added to commit (use "git add" and/or "git commit -a")

```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master|MERGING)
$ git add -A
```

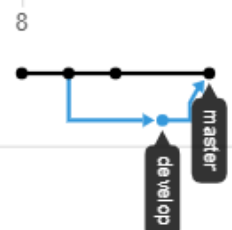
```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master|MERGING)
$ git commit -m "Fix conflict"
[master 9f50c67] Fix conflict
```

```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$ git push origin master
Enumerating objects: 1, done.
Counting objects: 100% (1/1), done.
Writing objects: 100% (1/1), 225 bytes | 225.00 KiB/s, done.
Total 1 (delta 0), reused 0 (delta 0)
To https://github.com/datnvhust/git.git
4a6cbe7..9f50c67  master -> master
```

```
datnv@DESKTOP-M6TJDS8 MINGW64 ~/Desktop/Git (master)
$
```

Network graph

Timeline of the most recent commits to this repository and its network ordered by most recently pushed to.

Owners	Sep
datnvhust	<div>8</div> 

🔑 master ▾

🔑 2 branches

🏷 0 tags

Go to file

Add file ▾

📄 Code ▾

🔗 Nguyễn Văn Đạt Fix conflict		9f50c67 3 minutes ago	🕒 5 commits
📄 README.md	Init	18 minutes ago	
📄 index.html	Change index.html in develop	7 minutes ago	

README.md

##Init

✎

6. Deploy server dùng git *

1. Login vào server

```
ssh your_user@server_ip_address // kết nối với server
```

2. Cài đặt git

```
sudo apt-get update // cập nhật apt-get  
sudo apt-get install git // cài đặt git
```

3. Khởi tạo git trong project

```
git clone url // Lấy code từ url
```

4. Chạy chương trình

CI/CD *

- Continuous Integration/ Continuous Delivery
- Quá trình tích hợp thường xuyên + cập nhật phiên bản mới

Tài liệu tham khảo

- <https://git-scm.com/downloads>
- Git Flow