**CHƯƠNG II: CÀI ĐẶT GIT**

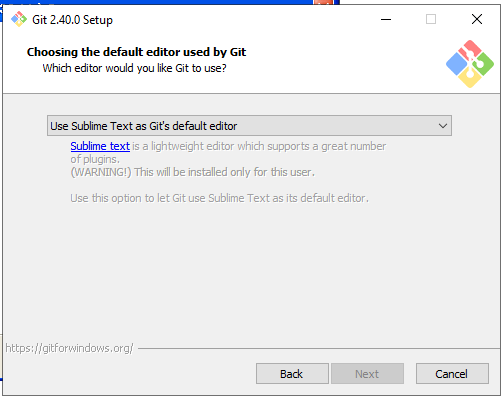
1. **CÀI ĐẶT GIT TRÊN WINDOWS**

Windows vốn là hệ điều hành quen thuộc với đại đa số người dùng.

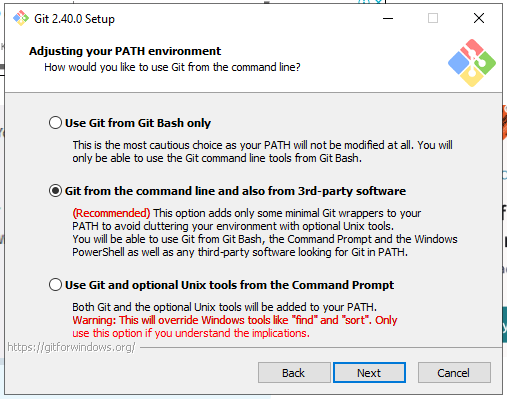
* Bước 1: Truy cập vào đường dẫn: [**https://gitforwindows.org/**](https://gitforwindows.org/)và sau đó ấn vào chữ Download.



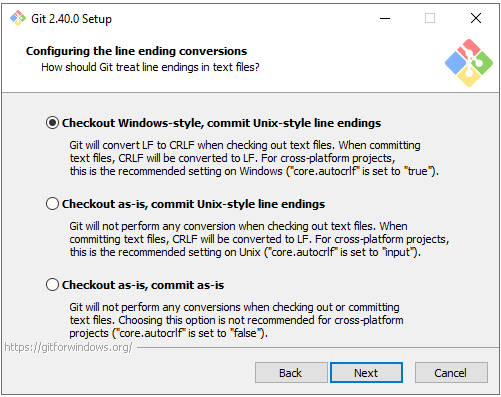
* Bước 2: Sau khi đã tải xuống thành công, click vào file để tiến hành cài đặt.
* Bước 3: Cài đặt bình thường như các phần mềm khác, về các options thì dựa vào yêu cầu của mỗi người để có những lựa chọn phù hợp.
* Bước 4: Cài đặt Git Editor cho Windows, có thể là các text editor như Vin, Notepad++, Sublime Text,…



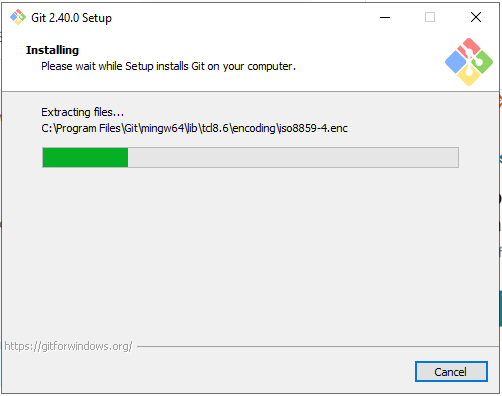
* Bước 5: Lựa chọn PATH môi trường phù hợp với nhu cầu.



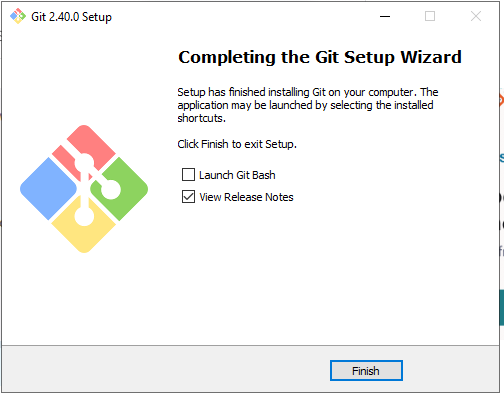
* Bước 6: Chọn phong cách dòng lệnh.



* Bước 7: Bấm Install để cài đặt.



* Bước 8: Kết thúc sau khi đã cài đặt thành công.



1. **CÀI ĐẶT GIT TRÊN LINUX**

Hệ điều hành Linux hỗ trợ Git rất tốt, đặc biệt là với công cụ Terminal mặc định của Linux, giúp Git phát huy được tối đa sức mạnh vốn có. Git sẽ có nhiều cách cài đặt khác nhau dựa trên từng Distro từ nhân Linux. Người dùng truy cập vào đường dẫn: <https://git-scm.com/download/linux> và làm theo đường dẫn sau.

* **Bước 1:** Mở Terminal và gõ 2 lệnh sau.



* **Bước 2:** Gõ lệnh ‘**git-version**’ để kiểm tra đã cài đặt thành công hay chưa.



**CHƯƠNG III: CẤU HÌNH**

1. **CẤU HÌNH GITHUB**
   1. **Tạo Github Repository**

Để làm việc, trước hết cần tạo một kho lưu trữ mã nguồn lệnh cho dự án, việc này về bản chất cũng giống như thao tác trên các dòng lệnh Git. Ở đây, người dùng sẽ làm qua giao diện với các cú nhấp chuột. Các bước tạo Repo ban đầu được mô tả như sau:

* **Bước 1:** Người dùng bấm vào ký hiệu dấu cộng been cạnh ảnh đại diện của tài khoản và chọn New Repository, Github sẽ chuyển qua trang khởi tạo Repo mới.

Graphical user interface, text, application, chat or text message

Description automatically generated

*Hình 3.1: Khởi tạo Github Repository.*

* **Bước 2:** Đặt tên Repo cũng như miêu tả dự án cần thiết.

Graphical user interface, text, application, chat or text message

Description automatically generated

*Hình 3.2: Đặt tên cho Repo.*



*Hình 3.3: Mô tả Repo dự án.*

* **Bước 3:** Chọn một trong hai tùy chọn do Github đưa ra. Hai tùy chọn có nội dung như sau:
  + Hai chế độ phân quyền là Public và Private. Public là chế độ mặc định, cho phép bất cứu ai cũng có thể xem được Repo. Nếu người tạo không muốn công khai có thể chuyển sang chế độ Private.

A screenshot of a computer

Description automatically generated

*Hình 3.4: Hai chế độ phân quyền của Repo Github.*

* + Tập tin README giới thiệu Repo kèm một tệp tin .gitgnore. Đây là một template có sẵn trong Github và cho mọi người tùy chọn.

Graphical user interface, text, application, chat or text message

Description automatically generated

*Hình 3.5: Template tùy chọn Repo Github.*

* **Bước 4:** Bấm nút Creat repository tạo Repo, Github sẽ chuyển trang Repo đã hoàn thành.

Graphical user interface, text, application, email

Description automatically generated

*Hình 3.6: Repo sau khi khởi tạo.*

Trong trưởng hợp người dùng không thêm tùy chọn, Github sẽ mặc định đây là người dùng mới với dịch vụ nó sẽ chuyển dến một trang hướng dẫn cho người dùng này.

Graphical user interface, text, application, email

Description automatically generated

*Hình 3.7: Trang hướng dẫn cho người dùng.*

* 1. **Xóa Github Repository**

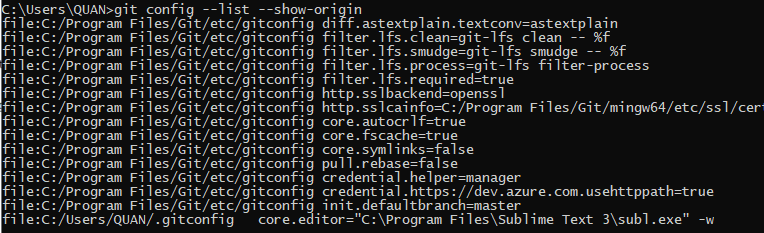
Nếu có nhu cầu xóa Repo thì yêu cầu tài khoản người dùng phải có đặc quyền hoặc là Admin cho một Repo của tổ chức. Các bước xóa Repo trên Github rất đơn giản nhưng có một vài điều cần lưu ý là tất cả các tệp đính kèm cùng phân quyền nhóm dự án đều bị xóa bỏ.

Người dùng có thể khôi phục lại hành động xóa Repo trong 90 ngày.

1. **CẤU HÌNH GIT CƠ BẢN**
   1. **Thiết lập ban đầu**

Điều đầu tiên cần làm sau khi cài đặt git là cấu hình chứng thực, thiết lập các tùy chọn ban đầu phù hợp với cá nhân người sử dụng.

Git có một số công cụ dòng lệnh là git config để người dùng tuy chỉnh các giá trị để git có giao diện và vận hành như họ muốn. Muốn xác định người dùng cần gõ lệnh git config ‘**–list –show-origin**’, nó sẽ hiện thị file config được lưu trên đường dẫn nào, cũng như các giá trị đã được cấu hình mặc định bởi Git.



*Hình 3.8: Xem thông số mặc định.*

* 1. **Danh tính của bạn**

Thông tin đầu tiên cần phải cấu hình trước khi sử dụng Git là tên và địa chỉ email của bạn. Git sẽ sử dụng hai thông tin này để gắn vào mỗi lệnh commit (mục đích để biết ai đã thực hiện commit).

Nên cấu hình danh tính của bạn ở mức global, và chỉ làm một lần. Git sẽ sử dụng thông tin này cho tất cả các thao tác bạn làm trên hệ thống. Nếu muốn sử dụng tên và email khác cho một dự án riêng, thì thực hiện thêm việc cấu hình danh tính tại mức local cho chính dự án đó.

Ví dụ: *$ git config --global user.name "Nam Cong"*

*$ git config --global user.email langyentrach@gmail.com*Mở **.gitconfig** để kiểm tra thông tin vừa cấu hình:

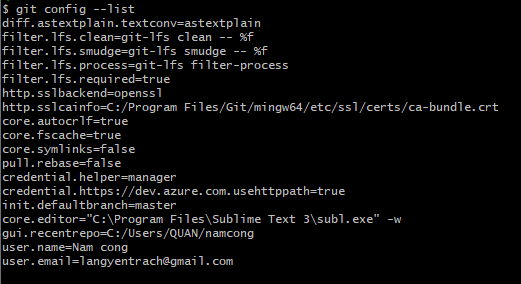
*[user]*

*name = Nam Cong*

*email = langyentrach@gmail.com*

* 1. **Kiểm tra cài đặt**

Git sẽ đọc một giá trị đến từ các file khác nhau, người dùng sẽ gõ lệnh ‘**git config –list**’ để liệt kê các thiết lập mà Git tìm thấy.



*Hình 3.9: Xem thiết lập.*