**1. Giới thiệu về git.**

Git là phần mềm quản lý mã nguồn phân tán được phát triển bởi Linus Torvalds dành cho việc phát triển Linux kernel. Git là phần mềm mã mở được phân phối theo giấy phép công GPL2.Git là tên gọi của một hệ thống quản lí phiên bản phân tán phổ biến nhất hiện nay. Git nghĩa là hệ thống giúp mỗi máy tính có thể lưu trữ nhiều phiên bản khác nhau của một mã nguồn được nhân bản (clone) từ một kho chứa mã nguồn (repository), mỗi thay đổi vào mã nguồn trên máy tính sẽ có thể ủy thác (commit) rồi đưa lên máy chủ nơi đặt kho chứa chính. Và một máy tính khác (nếu họ có quyền truy cập) cũng có thể clone lại mã nguồn từ kho chứa hoặc clone lại một tập hợp các thay đổi mới nhất trên máy tính kia. Trong Git, thư mục làm việc trên máy tính gọi là working tree.

**2. Các khái niệm cơ bản trong git.**

**2.1. Repository.**

Hay được gọi tắt là Repo đơn giản là nơi chứa/cơ sở dữ liệu (database) tất cả những thông tin cần thiết để duy trì và quản lý các sửa đổi và lịch sử của dự án. Trong Repo có 2 cấu trúc dữ liệu chính là Object Store và Index. Tất cả dữ liệu của Repo đèu được chứa trong thư mục bạn đang làm việc dưới dạng folder ẩn có tên là.git (không có phần tên trước dấu chấm).

**2.2. Object store.**  
 Object store là trái tim của Git, nó chứa dữ liệu nguyên gốc (original data files), các file log ghi chép quá trình sửa đổi, tên người tạo file, ngày tháng và các thông tin khác.  
 Git có 4 loại object là:Blobs, Trees, Commits, Tags

* Blobs: là file nhị phân có thể chứa được mọi loại dữ liệu bất kể là dữ liệu của chương trình gì.
* Trees: lớp đại diện cho thông tin thư mục như thông tin định danh của blob, đường dẫn, chứa một ít metadata chứa thông tin cấu trúc và các thư mục nhỏ có trong thư mục đó.
* Commits: Chứa metadata có thông tin về mọi thứ như tên tác giả, người tải lên (commiter), ngày tải lên, thông tin log...
* Tags: đánh dấu cho dễ đọc. thấy vì 1 cái tên dài như là 9da581d910c9c4ac93557ca4859e767f5caf5169, chúng ta có thể tên tag là Ver-1.0- Alpha. Dễ nhớ và dễ sử dụng hơn.

**2.2. Index.**

Index là file nhị phân động và tạm thời miêu tả cấu trúc thư mục của toàn bộ Repo và trạng thái của dự án được thể hiện thông qua commit và tree tại một thời điểm nào đó trong lịch sử phát triển. Git là một hệ thống truy tìm nội dung (content tracking system). Index không hứa nội dung file mà chỉ dùng để truy tìm (track) những thứ mà bạn muốn commit.

**3. Chức năng của git.**

Git giúp nhiều người có thể làm việc cùng nhau trên cùng một project, mỗi người có một kho code riêng để làm việc và có một kho code trung tâm thực sự nơi chứa các kết quả ổn định sau giai đoạn phát triển. Mỗi người làm việc trên một bản sao từ master, bạn có thể làm việc mọi thao tác trên local repository, sau khi khi hoàn thành công việc thì đẩy lên remote repository

**4. Các lệnh cơ bản về git.**

\_ git init : Dùng để tạo 1 repository mới.

\_ git clone /đường-dẫn-đến/repository/ : clone 1 repository có sẵn ở trên máy cục bộ

\_ git clone tênusername@địachỉmáychủ:/đường-dẫn-đến/repository : clone 1 repository ở máy chủ khác.

\_ git add <tên tập tin> : add file vào trạng thái state, tức là trạng thái được index, đã được theo dõi và sẵn sàng để đưa vào local repository.

\_ git commit : thêm các file ở trạng thái stage vào local repository sẵn sàng để đưa lên remote repository

\_ git checkout -b <tên nhánh> : tạo nhánh tại local để bắt đầu làm việc.

\_ git checkout <tên nhánh> : chuyển nhánh làm việc hiện tại tới nhánh này.

\_ git status : xem trạng thái của file hiện tại bao gồm

\_ git push origin <tên nhánh> : đẩy local repository lên server repository tại nhánh đó.

\_ git pull origin <tên nhánh> : cập nhật từ server xuống nhánh hiện tại trong local repository.

**5. Workflow với git.**

Thư mục cục bộ bao gồm 3 cây được duy trì bởi git. Đầu tiên là thư mục làm việc (Working Directory) chứa các tập tin hiện tại, cái thứ 2 là chỉ mục (index) đóng vai trò như staging area hay là những file đã được theo dõi và cuối cùng là HEAD trỏ đến commit gần đây nhất.

* Đầu tiên chúng ta sẽ clone nhanh làm việc của mình từ remote repository.
* Tiến hành làm việc trên nhánh của mình.
* Sử dụng lệnh git add để thêm các file vào staging area.
* Sử dụng lệnh git commit để thêm các file vào trạng thái local repository.
* Sử dụng lệnh git push origin <tên nhánh> để đẩy lên nhánh tại remote repository.
* Sử dụng lệnh git pull origin <tên nhánh> để lấy các cập nhập tử remote repository về local.