* $ git config –global user.name” [Tenuser]”: Khai báo tên đăng nhập
* $ git config –global user.email”[Email]”: Khai báo email
* $ git config –global color.ui.auto : Tô màu kết quả xuất
* $ git init [TenProject]: Tạo 1 git repository 1 project mới hoặc đã có
* $ git clone [url]: Coppy bản sao của dự án
* $ git status :Kiểm tra trang thái của các file đã thay đổi trong thư mục làm việc
* $ git diff: Hiển thị thông tin thay đổi giữa thư mục làm việc và vùng index [staging] hoặc với commit cũ, thong tin thay đổi giữa index và commit , thông tin thay đổi giũa 2 nhánh
* $ git add [file] : Thêm mục thay đổi đến file trong thư mục
* $ diff –staged : Kiểm tra sự thay đổi của index với commit cuối
* $ git reset[file]: hủy commit
* $ git commit-m”[nội dung ]” Ghi lại thông tin
* $ git branch: liệt kê các nhánh
* $ git branch [ ten nhanh] thêm một nhánh
* $ git checkout [ten nhanh]: chuyển sang nhánh khác
* $ git merge [nhánh] :Kết hợp branch
* $ git branch –d [ten nhanh] : xóa branch
* $ git rm [file]: xóa file
* $ git rm –cached[file]: gỡ bỏ file
* $ git mv [ten file ban dau] [ten file muon doi] : đổi tên file và commit

\*.log

Build/

Temp-\* 1 văn bản ngăn chặn việc tạo các phiên bản ngẩu nhiên của file

* $ git ls-files-other-ignored-exclude-standard:Các file bị bỏ qua trong dự án
* $ git stash : lưu trạng thái chưa commit
* $ git stash pop: quay lại thay đổi lưu trong stack
* $ git stash list : xem danh sách các lần thay đổi
* $ git stash drop : xóa nội dung đã lưu trong stack
* $ git log: xem lại thông tin về lịch sử đã commit
* $ git log –follow[file] :Liệt kê các phiên bản
* $ git diff [nhanh dau tien]…[nhanh thu 2]: Kiểm tra thay đổi giữa 2 branch với nhau
* $ git show [commit]: xem lại commit
* $ git reset [commit] :quay lại các phiên bản commit trước
* $ git reset –bard[commit]: Hủy lịch sử
* $ git fetch [bookmark]: Không hợp nhất những thay đổi
* $ git merge [bookmark]/[branch]:Hợp nhánh
* $ git push[alias][brach]: đăng branch
* $ git pull: tìm nạp và tải xuống