**RANGKUMAN GITHUB**

Git adalah sebuah version control yang digunakan untuk berkolaborasi antara programmer / developer dalam mengatur source code dari sebuah aplikasi /program. Git dibuat oleh Linus Torvalds dan komunitas pengembangan linux, berbeda dengan version control yang lain seperti subversion, git merupakan distributed version control. Artinya, repository git berada terpencar-pencar dan user bisa melakukan commit dan rollback langsung pada repository lokal sendiri, kemudian hasil commit/rollback tersebut bisa dipush ke repository server agar developer/programmer lain bisa mengaksesnya (checkout).

# Membuat Sebuah Repository

Untuk membuat repository baru caranya cukup mudah. Pada Windows, cukup klik kanan di dalam folder yang ingin dijadikan repository, kemudian pilih ‘Git GUI Here’ atau ‘Git Bash Here’.

Untuk membuat sebuah repository, buat clone dari repository yang telah ada. Clone artinya anda meng-copy semua file yang ada di repository tersebut dan menjadikannya repository lokal anda. Kemudian, lakukan commit terhadap perubahan yang dilakukan ke repository lokal, kemudian melakukan push ke server. Berikut adalah contoh perintah untuk melakukan clone :

git clone https://username\_anda@bitbucket.org/username\_anda/repository\_anda.git

Diatas adalah contoh bagaimana meng-clone sebuah repository dengan cara menjalankan git clone kemudian diikuti oleh URL dari repository yang diinginkan. Setelah menjalankan perintah tersebut, tunggu proses cloning hingga selesai,. Sesudah itu masukkan credential(username/password) dari repository yang barusan di clone.

Selain dengan menggunakan perintah seperti diatas, anda juga bisa menggunakan GUI. Bagi pengguna windows, klik kanan kemudian pilih ‘Git Gui’ pada sebuah folder baru yang masih kosong. Kemudian, pilih clone dan masukan URL dari repository yang diinginkan. Bagi pengguna Linux, bisa menggunakan aplikasi seperti RabbitVCS yang menyediakan GUI baik untuk Git, Svn dan Mercurial.

# Staging

Untuk melakukan commit, terlebih dahulu harus melakukan staging terhadap file yang akan dicommit tersebut. Fungsi dari stagging ini salah satunya yaitu untuk memisahkan antara file yang akan dicommit dan tidak akan dicommit dimana hanya file sudah di stages lah yang bisa di-commit. Untuk melakukan stagging lakukan perintah dengan contoh sebagai berikut:

$ git add index.html

Dimana ‘index.html’ merupakan nama file yang akan di stages. Selain dengan menggunakan perintah seperti diatas, anda juga bisa melakukan stagging melalui GUI dengan melakukan klik kanan pada folder repository, kemudian pilih ‘Git Gui’. Pada windows yang muncul anda akan melihat sebuah tombol bertuliskan ‘stages change’. Kelebihan menggunakan GUI adalah dapat melakukan stagging terhadap banyak file hanya dengan menggunakan satu kali klik.

# Commit

Untuk melakukan commit caranya mudah saja, gunakan perintah sebagai berikut:

git commit -m "pesan commit"

Dengan menggunakan perintah seperti diatas, maka semua file yang berada di stagging area ( file yang sudah di-stages ) akan di-commit  ke dalam repository, dimana ‘pesan commit’ wajib diisi untuk mendeskripsikan perubahan yang anda lakukan. Misalnya, jika anda melakukan perubahan pada file ‘index.html’ maka ‘pesan commit’ tersebut bisa anda buat misalnya sebagai berikut:

git commit -m "perubahan pada file index.html"

Seperti halnya melakukan stagging, juga bisa melakukan commit dengan menggunakan Git Gui, hal ini terutama berlaku jika menggunakan Windows sebagai OS pilihan. Jika menggunakan linux, dapat menggunakan aplikasi pihak ketiga seperti RabbitVCS.

# Push

Fungsi push digunakan untuk mengirim data ke repository lain. Misalnya menyimpan source code yang telah dirubah , user dapat menggunakan perintah git push:.

git push ../remote-repository.git

# Pull

Fungsi pull digunakan untuk mengambil update terbaru dari repository lain. Misalnya pada repository B terdapat perubahan data terhadap sebuah souce code maka user dapat mengambil data terbaru dengan perintah git pull :

git pull ../remote-repository.git

# Merge

Fungsi merge digunakan untuk menggabungkan source code yang berada di server dan source code yang ada di local dapat digunakan fungsi git merge:

git merge testing

# Revert

Fungsi revert digunakan untuk mengembalikan commit ke status sebelumnya. Misalnya melakukan perubahan pada sebuah file bernama ‘index.html’. Jika melakukan kesalahan pada perubahan tersebut. bisa mengembalikannya ke status sebelumnya dengan menggunakan fungsi revert:

git revert <commit>

Dimana <commit> adalah index dari commit tujuan dimana kita akan melakukan revert. Perlu diketahui bahwa revert ini merupakan cara mengembalikan commit ke versi sebelumnya yang paling aman. Selain revert, ada juga fungsi reset pada git yang lebih beresiko, karena memungkinkan beberapa perubahan / file tertentu hilang dari repository.