# **Tugas Resume Tengku mahmudi**

1. Introduction and instalationn

Pertama introduction dimulai dengan pengenalan apa itu Versioning yaitu mengatur versi dari source kode program. Melakukan setting up , clone dan config ,Menyimpan perubahan , menambah , mengambil , mengahapus ,berpindah, kembali dan mengabaikan dengan beberapa coommandline.

Git merupakan version control system, mengelola perubahan dari sesbuah dokumen program komputer website dan kumpulan infromasi lainnya.

~ sebuah sistem yang menyimpan rekaman peribuhana pada source code.

~ memungkinkan bekerja berkolaborasi dengan lebih baik.

~ mengtahui siapa yang melakukan dan kapak sebuah prubahan terjadi.

~ memungkinkan kita untuk kembali ke keadaan sebelum perubahan (checkout)

aplikasi selain git : subversion ,mercurial,vcs

~ git adalah sebuah vcs terdistribusi untuk mengelola perubahan file didalam folder.

~ sebuah software untuk mengelola perubahan file didalam folder (repository/repo).

~ riwayat perubahan file disimpan menggunakan serangkaian commit.

~ harus repositori agar git bisa memantau file

~ setelah membuat file harus commit dulu

~ git repo bisa membuat cabang dalam 1 repo yang sama (brach / cabang)

Kedua instatation bisa dilakukan dengan mendownload melalui link :

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

1. Setting up and Push

Diawali dengan membuat email dan nama melalui aplikasi Git bash yang kita download sebelumnta dengan comandline :

git config --global user.email "mudibae997@gmail.com"

git config --global user.name "mudi"

~ kemudian mengecek status,menambahkan dan memberi komen setiap perubahan yang kita lakukan dengan commandline :  
git add . memasukan semua directory distaging area

git commit -m "inisialisasi index.html" \*message harus jelasss

git push origin , main default git

git add <directory>

git add hello.py

~ kemudian membahas commandline

git diff --staged , mengetahui bagaimana perubahan yang diedit

git stash , menyimpan perubahan , dapat dikembalikan berubahan

git stash apply

stash bisa menghapus

pop stash mengembalikan

~ apa itu gitigrnore

.gitignore berfungsi untuk memfilter mana aja yang bisa masuk kerepository dan tidak

~ Kemudian mengecek log and checkout dengan command line :

git log --oneline ,untuk menampilkan setiap commit yang dilakukan

git checkout (kode) bisa untuk pindah branch

git reset (kode) --soft (perubahan tidak terjadi sebanyak hard , commit tetap ada )

git reset (kode) --hard ( bisa mengapus commit tidak bisa kembali atau diakses lagi)

~kemudian mencoba comand git push,fetch dan pull

git remote -v

git remote add origin https....

git fetech

git pull origin main (kembali ke repositori main)

git push origin main (memasukan perubahan ke repository)

1. Branch and conflic

~ kemudian mencoba command git brancing dan git merge

BRANCH cabang atau melindungi main project

git bracnh --list ,mengetahui list atau branch apa aja yg udah dibuat

git branch <branch> , membuat branch baru

git branch -D <branch> ,mendelete

git branch -a , remote

git checkout -b new-feature master

git add <file>

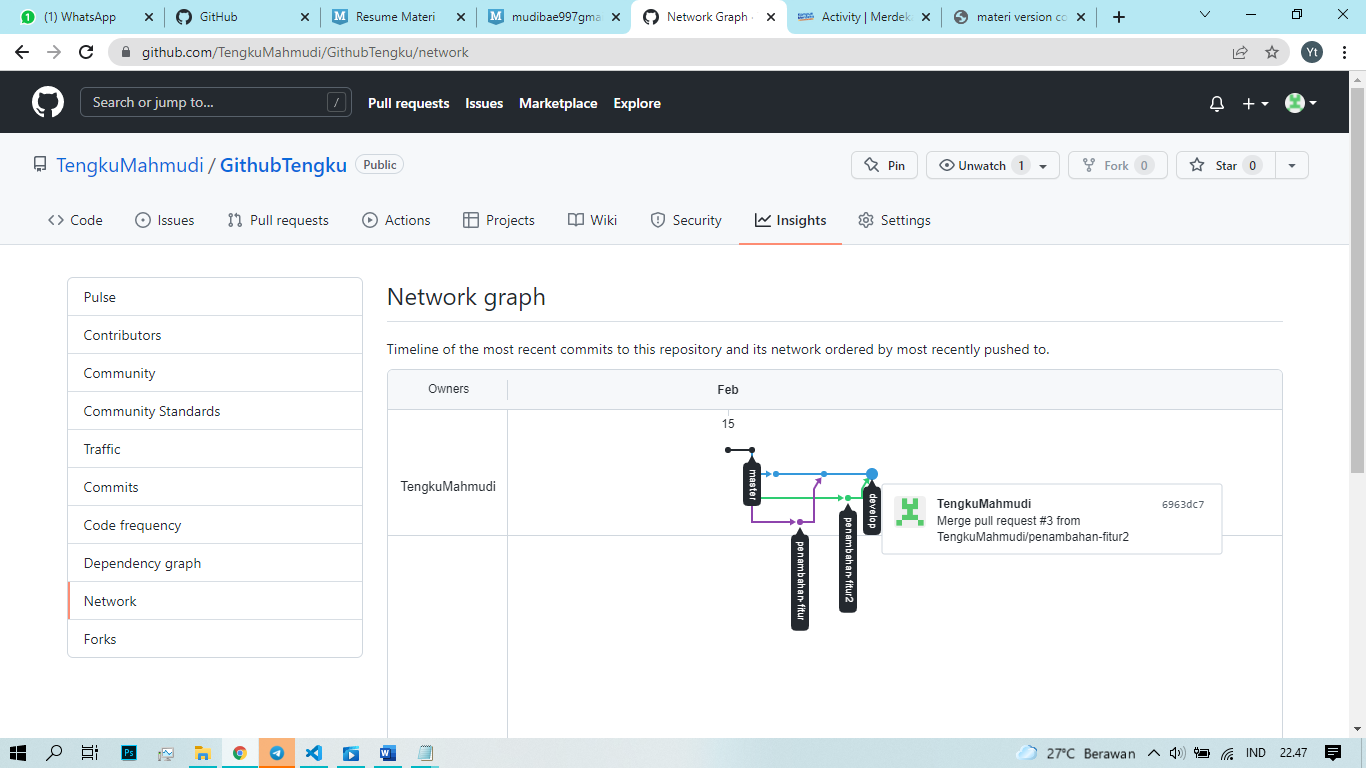
git commite -m “start a feature”

git checout master

git merge new-feature(nama)

git push -u origin develop(nama branch)

~ kemudian kita mencoba meminta pull request kepada seseorang atau perusahaan untuk merubah file pekerjaan yang mengalami kesalahan melalui web github, selanjut nya jika kita ingin melakukan pekerjaan workflow dengan optimal kita harus membuat main branch atau master tidak terganggu. Untuk merubah atau menambah master kita perlu membuat branch fitur baru diluar dari master agar tidak merubah fungsi awal dari master. Jika memang yakin sudah sesuai dengan fitur fitur yang disepakati bisa kita lakukan merge dari branch fitur ke master seperti gambar dibawah ini :



Fungsi master tidak berubah tetapi branch develop yang melakukan merging dengan perubahan fitur dan jika branch develop sudah sempurna dapat kita gabungkan ke master.

\*tambahan

repo : folder project

commit : rekaman/snapshot pada perubahan tertentu

hash : penanda unik pada sebuah commit

chechout : berpindahan kesebuah commit

branch : cabang bebas dari sebuah commit

merge : menggabungkan branch

remote : sumber yang memiliki repository

clone : mengambil repo dari remote

push : mengirimkan commit dari repository

pull : mengambil commit dari repository