

PENGUMPULAN TUGAS DENGAN GIT

Oleh: Dewi Anisa Istiqomah

A. Pengaturan Awal Git

1. Unduh git sesuai dengan OS yang digunakan di git-scm.com.
2. Install git. Untuk memastikan git sudah terinstall yaitu dengan mengetikkan command **git** pada terminal.
3. Buat akun pada github.com.
4. Lakukan konfigurasi (<https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup>) untuk memberitahukan identitas yang sudah didaftarkan pada github.com dengan menuliskan command berikut pada terminal:
 1. **git config --global user.name "Dewi Anisa Istiqomah"**. Global, karena akan menggunakan secara keseluruhan sistem.
 2. **git config --global user.email "dewianisaist@gmail.com"**
5. Cek apakah konfigurasi sudah berhasil dengan menuliskan command **git config --list**. Jika sudah berhasil maka akan tertera nama dan email user.

B. Cara Pengumpulan Tugas dengan Git

- Misal akun github mahasiswa adalah Dewi Anisa Istiqomah, yang mana akun ini akan digunakan untuk mengumpulkan tugas. Dan akun instruktur adalah KL Web Lanjut, yang mana akun ini akan berisikan tugas-tugas yang harus dikerjakan.
- Sebagai contoh: Mahasiswa akan mengumpulkan tugas 02 yaitu tentang analisis kebutuhan proyek akhir dalam repositori tugas02-analisis-kebutuhan-pa di akun github instruktur. Dalam repositori tersebut, instruktur telah menginisialisasi terlebih dahulu.
- Berikut ini langkah-langkah untuk pengumpulan tugas:
 1. **Buka repositori tugas** yang dimaksud pada github instruktur.
 2. **Fork repositori** tersebut ke akun github masing-masing mahasiswa dengan cara **klik button Fork di kanan atas**. Fork bertujuan untuk meng-copy repositori agar tidak tercampur dengan repositori asli milik instruktur.
 3. **Clone repositori dengan cara copy to clipboard untuk url HTTPS**. Clone digunakan untuk menjalankan repositori di lokal.
 4. Buka terminal. **Akses folder untuk menyimpan repositori yang akan dijalankan di lokal** dengan command **cd**.
 5. Lakukan clone dengan command **git clone url-yang-telah-dicopy**.
 6. **Cek apakah repositori telah berhasil diclone** dengan command **ls**. Jika sudah berhasil, maka muncul folder tugas02-analisis-kebutuhan-pa. **Akses folder tersebut** dengan perintah **cd tugas02-analisis-kebutuhan-pa**.

7. **Cek remote repositori apakah sudah terdaftar** dengan command ***git remote -v***.
Jika sudah terdaftar akan muncul seperti berikut:

```
Dewis-Air:tugas02-analisis-kebutuhan-pa dewianisaist$ git remote -v
origin https://github.com/weblanjutabc2017/tugas02-analisis-kebutuhan-pa.git (fetch)
origin https://github.com/weblanjutabc2017/tugas02-analisis-kebutuhan-pa.git (push)
```

8. **Kerjakan tugas** sesuai dengan perintah di soal.
9. **Simpan file tugas** di folder repositori ybs. Misal: Contoh.docx.
10. Lakukan **cek apakah di dalam repositori ada perubahan**, dapat dengan mengetikkan command ***git status***. Contoh.docx berwarna merah, artinya file tersebut **untracked**.
11. Untuk **melakukan tracked** terhadap file yang untracked, dengan mengetikkan command ***git add Contoh.docx***. **Note:** Untuk melakukan tracked secara interaktif dapat dengan command ***git add -i***. **Interaktif** maksudnya adalah kita bisa memilih mana file yang akan dicommit duluan atau yang akan dicommit belakangan. Jika sudah pernah melakukan tracked terhadap suatu file, maka pilih update kemudian pilih file yang diupdate. Jika file belum pernah ditracked maka pilih add untracked, kemudian pilih file yang akan ditracked. **Important: File yang ditracked adalah file yang akan dilakukan commit.**
12. **Cek status lagi** dengan command ***git status***. Contoh.docx sudah berwarna hijau, artinya file tersebut tracked. Tracked berarti file sudah ditambahkan ke dalam staging area, yang mana siap untuk dicommit.
13. **Lakukan commit** untuk memberikan catatan terhadap perubahan pada repositori dengan command ***git commit -m "Isi pesan"***. Isi pesan sebaiknya spesifik sesuai dengan perubahan yang anda lakukan agar mudah untuk dipahami yang lain. Misal: ***git commit -m "Menambahkan file tugas 02"***. **Note:** setiap kali melakukan perubahan, langsung dilakukan commit, tidak perlu menunggu perubahan besar.
14. **Lakukan push**, gunakan command ***git push -u origin master***. Command ***-u*** hanya digunakan jika anda baru pertama kali melakukan push. Untuk push kedua dan seterusnya cukup ketikkan command ***git push origin master***. Biasanya anda diminta untuk memasukkan username dan password github, maka isikan sesuai permintaan.
- ***git push*** digunakan untuk melakukan push ke server tujuan.
 - ***origin*** adalah nama alias server yang diperoleh dari informasi ***git remote -v***.
 - ***master*** adalah branch. Branch tujuannya untuk memisahkan pekerjaan/code base yang ada. Jadi kedudukan di branch master bisa berbeda dengan branch yang lain tergantung kita commit di branch apa.
15. **Cek di github**, apakah file sudah tercatat.

16. Untuk melakukan pengumpulan ke instruktur maka lakukan **push request**. Klik **button New pull request** di github. Pull request harus dilakukan supaya instruktur dapat mengetahui bahwa mahasiswa X sudah menyelesaikan tugas dan siap untuk dinilai.
17. Akan muncul komparasi antara repositori milik instruktur dan mahasiswa. Dan dari komparasi itu terlihat adanya perbedaan dari kedua repositori. Kemudian klik **button Create pull request** untuk pengumpulan tugas.
18. **Isikan judul**, sesuai dengan tugas yang dikumpulkan.
19. **Isikan comment** dengan isian NIM, Nama dan Kelas.
20. **Klik button Create pull request**.
21. Di daftar pull request milik instruktur akan muncul siapa saja yang sudah mengumpulkan.
22. Sudah selesai, tinggal dilakukan penilaian oleh instruktur.

Note:

- Cara di atas yaitu menggunakan terminal. Jika ingin menggunakan tools berbasis GUI dapat menggunakan SourceTree atau tools lainnya.
- Akun github yang disebutkan di atas maksudnya adalah akun masing-masing kelas KL Pemrograman Web Lanjut, bukan akun pribadi instruktur.