



Modul Praktikum Algoritma Pemrograman Dasar



“Git & Github”

11.1. Tujuan

- Mahasiswa mampu memahami kegunaan dari Git dalam versi kontrol proyek
- Mahasiswa mampu mengelola repository dengan menggunakan Git
- Mahasiswa mampu menghubungkan repository lokal dengan Github
- Mahasiswa mampu menggunakan perintah Git untuk melakukan commit, push, dan pull

11.2. Pengertian Git

Git adalah sistem kontrol versi yang membantu kita melacak perubahan pada kode atau proyek kita. Bayangkan kita sedang menulis kode panjang, dan setiap kali kita menambahkan baris baru atau mengedit kode, kita menyimpan versi baru dari file tersebut tanpa menghapus versi sebelumnya. Dengan Git, kita bisa melihat seluruh riwayat perubahan yang pernah kita buat, dan jika perlu, kita dapat kembali ke versi sebelumnya tanpa kehilangan pekerjaan yang sudah kita lakukan.

11.3. Pengertian Github

Github adalah layanan berbasis cloud yang memungkinkan kita menyimpan repository Git secara online. Bayangkan Github adalah media sosial untuk kode, kita bisa menyimpan proyek kita, berbagi dengan orang lain, dan bekerja sama dalam sebuah tim. Github juga menyediakan berbagai fitur tambahan seperti pelaporan bug, manajemen tugas, dan integrasi dengan layanan lain yang mempermudah pengembangan perangkat lunak.

11.4. Manfaat Menggunakan Git dan Github

- Kolaborasi bekerja bersama tanpa khawatir saling menimpa pekerjaan.
- Backup online yang menyimpan kode kita di cloud.
- Pengelolaan versi dengan Git.



11.5. Konsep Dasar Git

- **Repository (Repo)**

Tempat penyimpanan semua file proyek dan riwayat perubahan. Repo bisa berada di komputer lokal (lokal repo) atau di server (remote repo).

- **Commit**

Tindakan menyimpan snapshot dari proyek kita. Setiap kali kita membuat commit, kita menyimpan riwayat perubahan yang telah kita lakukan.

- **Branch**

Cabang dari proyek utama yang memungkinkan kita bekerja pada fitur atau perbaikan tanpa mengganggu versi stabil dari proyek. Cabang utama sering disebut sebagai ``main`` atau ``master``.

- **Push**

Setelah menyimpan perubahan secara lokal, kita bisa mengunggahnya ke Github agar bisa diakses dari mana saja.

- **Merge**

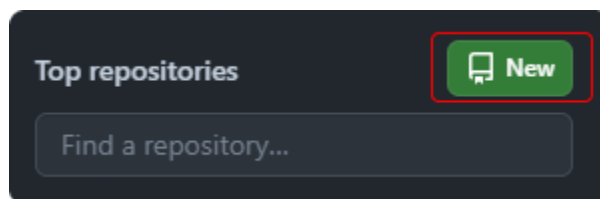
Proses menggabungkan perubahan dari satu branch ke branch lain.

- **Pull Request**

Permintaan untuk menggabungkan perubahan dari satu branch ke branch lain, biasanya digunakan dalam kerja tim sebelum perubahan diterapkan ke branch utama.

11.6. Membuat Repository di Github

1. Pastikan sudah membuat akun di Github.
2. Buat repository baru:
 - Klik tombol “New Repository” di halaman Github.



ALGORITMA PEMROGRAMAN DASAR / INFORMATIKA UNMUL	



- Isi nama repository, kemudian isi deskripsi repository jika diperlukan, pilih sifat repository public atau private.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*

Owner *

Repository name *

MatchaTi

my-repository

Nama Repository

✔ my-repository is available.

Great repository names are short and memorable. Need inspiration? How about [verbose-spoon](#) ?

Description (optional)

Repository Pertama Saya

Deskripsi Repository (boleh diisi boleh tidak diisi)

☒ Public

Bisa dilihat oleh siapapun

Anyone on the internet can see this repository. You choose who can commit.

☐ Private

Hanya bisa dilihat oleh orang tertentu

You choose who can see and commit to this repository.

Initialize this repository with:

☒ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: None

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

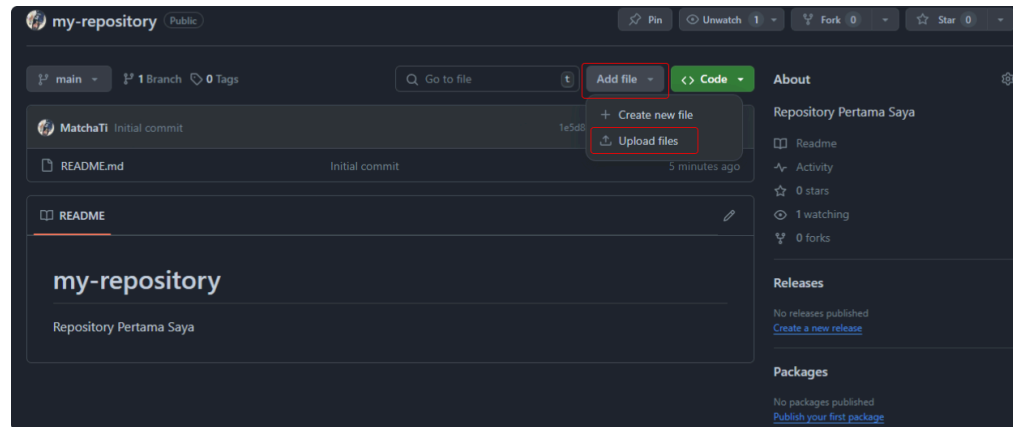
This will set `main` as the default branch. Change the default name in your [settings](#).

☐ You are creating a public repository in your personal account.

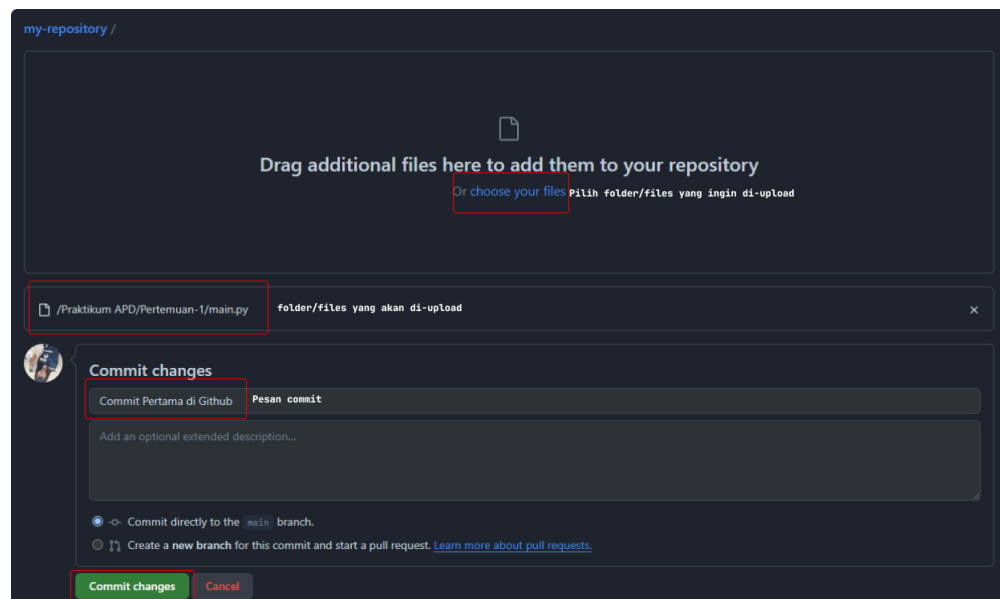
Create repository



- Klik “Create repository”
3. Upload folder/files ke Github.
- Klik tombol “Add file”, kemudian pilih “Upload files”.



- Drag file yang ingin di-upload dan isi pesan commit.



- Kemudian klik “Commit changes”.



4. Tampilan ketika berhasil upload folder/files di Github.

