

GitHub (Penjelasan dan Fungsional)

Unit Kopetensi : J.620100.023.02



Jurusan Web Programming

Jl. Karet Pasar Baru Barat Nomor 23, Kelurahan Karet Tengsin, Kecamatan Tanah
Abang, Kota Administrasi Jakarta Pusat, 10250

Apa Itu GitHub?

GitHub adalah platform berbasis cloud yang digunakan untuk menyimpan, mengelola, dan berbagi kode dengan menggunakan sistem kontrol versi **Git**. Dengan GitHub, pengembang dapat berkolaborasi dalam pengembangan perangkat lunak dengan lebih mudah.

Fungsi Utama GitHub

1. Menyimpan Kode

- GitHub berfungsi sebagai tempat penyimpanan proyek berbasis Git, baik secara **public** (terbuka untuk semua) maupun **private** (hanya bisa diakses oleh tim tertentu).

2. Kolaborasi dalam Tim

- Dengan GitHub, beberapa orang bisa bekerja dalam satu proyek yang sama tanpa takut kehilangan perubahan.

3. Menggunakan Kontrol Versi (Version Control)

- GitHub memungkinkan pengembang untuk melacak perubahan kode, kembali ke versi sebelumnya, dan mengelola beberapa versi kode secara efisien.

4. Membantu dalam Pengembangan Open Source

- Banyak proyek open-source tersedia di GitHub, yang memungkinkan siapa saja untuk berkontribusi atau mempelajari kode dari pengembang lain.

5. Menyediakan Fitur CI/CD

- GitHub mendukung **Continuous Integration dan Continuous Deployment (CI/CD)** untuk otomatisasi pengujian dan penerapan kode.

Komponen Utama di GitHub

1. Repository (Repo)

- Tempat penyimpanan kode dalam proyek.

2. Branch

- Versi cabang dari repository utama untuk mengembangkan fitur baru tanpa mengganggu kode utama.

3. Commit

- Catatan perubahan yang dilakukan pada kode.

4. Pull Request (PR)

- Permintaan untuk menggabungkan perubahan kode dari satu branch ke branch lainnya.

5. Issues

- Digunakan untuk melacak bug, fitur yang diusulkan, atau diskusi terkait proyek.

6. Fork

- Menyalin repository orang lain ke akun pribadi untuk melakukan perubahan tanpa mempengaruhi kode asli.

Perbedaan Git dan GitHub

Git	GitHub
Sistem kontrol versi (VCS) untuk mengelola perubahan kode	Layanan berbasis cloud yang menyimpan repository Git
Bisa digunakan secara offline	Harus terhubung ke internet untuk mengakses repository online
Berjalan di komputer lokal	Berfungsi sebagai remote repository untuk berbagi dan berkolaborasi
Perintah dijalankan melalui terminal/command prompt	Dapat digunakan melalui antarmuka web dan Git CLI

Kesimpulan

GitHub adalah platform yang sangat berguna bagi pengembang perangkat lunak, baik untuk proyek individu, tim, maupun open-source. Dengan fitur-fitur seperti **repository**, **commit**, **branch**, **pull request**, dan **issue tracking**, GitHub membantu mengelola proyek lebih efisien.

Pembuatan Akun dan Push Proyek

Berikut adalah langkah-langkah untuk membuat akun GitHub dan mengunggah (push) proyek web ke GitHub:

A. Membuat Akun GitHub

1. Buka Situs GitHub

- Kunjungi <https://github.com>.

2. Mendaftar Akun

- Klik tombol **Sign Up**.
- Masukkan **username**, **email**, dan **password**.
- Klik **Create account**.
- Verifikasi email melalui pesan yang dikirim ke email Anda.

3. Selesai!

- Setelah berhasil masuk, Anda siap untuk menggunakan GitHub.

B. Mengunggah Proyek Web ke GitHub

Berikut adalah langkah-langkah untuk menyimpan proyek ke GitHub menggunakan Git:

1. Install Git

- Jika belum terinstall, unduh dan pasang Git dari <https://git-scm.com/downloads>.
- Pastikan Git sudah terpasang dengan menjalankan perintah berikut di **Command Prompt (Windows)** atau **Terminal (Mac/Linux)**:

```
git --version
```

Jika Git terinstall, akan muncul versi Git.

2. Konfigurasi Git di Komputer

- Buka terminal/cmd, lalu jalankan:

```
git config --global user.name "NamaAnda"
```

```
git config --global user.email "emailanda@example.com"
```

Pastikan email yang digunakan sama dengan email GitHub.

3. Buat Repository di GitHub

- Masuk ke GitHub, klik tombol **New Repository** atau buka <https://github.com/new>.
- Isi nama repository (misalnya: project-web).
- Pilih **Public** atau **Private**.
- Klik **Create Repository**.

4. Inisialisasi Git di Proyek Web

- Buka terminal/cmd di dalam folder proyek web Anda.
- Jalankan perintah berikut:

```
git init
```

Ini akan membuat repository Git di dalam folder proyek Anda.

5. Hubungkan ke Repository GitHub

- Salin URL repository GitHub yang tadi dibuat.
- Jalankan perintah berikut di terminal/cmd:

```
git remote add origin https://github.com/username/project-web.git
```

6. Tambahkan dan Commit File Proyek

- Menambahkan semua file ke Git:

```
git add .
```

- Commit perubahan:

```
git commit -m "First commit"
```

7. Upload (Push) ke GitHub

- Jalankan perintah berikut untuk mengunggah proyek ke GitHub:

```
git branch -M main
```

```
git push -u origin main
```

- Jika diminta login, masukkan username dan token GitHub Anda (bisa dibuat di **Settings > Developer Settings > Personal Access Token** di GitHub).

8. Cek di GitHub

- Buka repository di GitHub dan pastikan semua file proyek telah terunggah.

Unggah File Baru ke Repository yang Sudah Ada

Jika Anda ingin mengunggah **file baru** ke repository GitHub yang sudah ada, ikuti langkah-langkah berikut:

1. Masuk ke Folder Proyek

Buka terminal atau **Command Prompt (cmd)** di dalam folder proyek yang sudah dikaitkan dengan repository GitHub.

2. Cek Status Git

Jalankan perintah berikut untuk melihat apakah ada file baru atau perubahan yang belum diunggah:

```
git status
```

Jika ada file baru yang belum ditambahkan ke Git, file tersebut akan muncul dalam daftar berwarna merah.

3. Tambahkan File Baru ke Git

Tambahkan semua file baru ke dalam Git dengan perintah:

```
git add .
```

Atau, jika hanya ingin menambahkan file tertentu, gunakan:

```
git add nama_file
```

Misalnya, jika Anda menambahkan file **index.html**, jalankan:

```
git add index.html
```


4. Commit Perubahan

Setelah file ditambahkan, buat commit dengan pesan deskriptif:

```
git commit -m "Menambahkan file index.html"
```

5. Upload (Push) ke GitHub

Unggah file baru ke repository GitHub dengan perintah:

```
git push origin main
```

Pastikan nama branch sesuai dengan branch utama repository Anda. Jika menggunakan master, ganti main dengan master:

```
git push origin master
```

6. Cek di GitHub

Buka repository GitHub Anda dan pastikan file baru sudah muncul di sana.

Catatan:

- Jika pertama kali menggunakan git push, Anda mungkin akan diminta memasukkan username dan token GitHub.
- Untuk mendapatkan token GitHub, buat di **Settings > Developer Settings > Personal Access Token** di GitHub.
- Jika repository Anda menggunakan branch selain main atau master, cek dengan perintah git branch dan gunakan nama branch yang sesuai.