

**Nama : Riski Jaya Putra**

**NPM : 2315061065**

**Kelas : PW-A**

---

### **SOAL**

1. Jelaskan apa yang dimaksud dengan Version Control System (VCS) dan sebutkan manfaatnya dalam pengembangan proyek web.
2. Sebutkan fungsi dan perbedaan dari perintah git init dan git clone dalam proses awal penggunaan Git.
3. Jelaskan apa yang dimaksud dengan branch dalam Git serta manfaatnya dalam mengembangkan fitur baru tanpa mengganggu branch utama.
4. Bagaimana cara kerja GitHub dalam mendukung kolaborasi antar-developer menggunakan konsep remote repository?
5. Uraikan alur kerja (workflow) sederhana Git dari tahap inisialisasi proyek hingga perubahan dikirim ke GitHub.

## **1. Pengertian Version Control System (VCS) dan Manfaatnya dalam Pengembangan Proyek Web**

Version Control System atau VCS adalah sistem yang berfungsi untuk mencatat dan mengelola setiap perubahan yang terjadi pada file atau kumpulan file dalam sebuah proyek. Dengan adanya VCS, seorang pengembang bisa melacak riwayat perubahan, membandingkan versi lama dan baru, serta memulihkan file jika terjadi kesalahan. Dalam pengembangan proyek web, sistem ini sangat membantu karena memungkinkan banyak orang bekerja secara bersamaan tanpa saling mengganggu hasil kerja satu sama lain.

Selain itu, VCS juga menjaga agar pekerjaan tetap aman karena semua versi tersimpan dengan baik. Pengembang tidak perlu takut kehilangan kode, sebab setiap perubahan terdokumentasi secara otomatis. Sistem ini juga membuat kolaborasi lebih efisien karena setiap anggota tim bisa fokus pada bagiannya masing-masing tanpa harus bekerja di file yang sama secara langsung.

## **2. Fungsi dan Perbedaan antara git init dan git clone**

Dua perintah dasar yang sering digunakan di awal kerja dengan Git adalah git init dan git clone, dan keduanya memiliki fungsi yang berbeda. Perintah git init digunakan untuk membuat repository baru dari nol di komputer lokal. Jadi, misalnya kamu punya folder kosong dan ingin menjadikannya proyek Git, cukup jalankan perintah ini dan Git akan menambahkan folder .git untuk mulai mencatat perubahan.

Sementara itu, git clone digunakan untuk menyalin repository yang sudah ada, biasanya dari GitHub atau server lain, ke komputer lokal. Dengan cara ini, kita tidak perlu membuat repository dari awal karena proyek dan riwayat commit-nya sudah tersedia. Secara sederhana, git init cocok digunakan ketika memulai proyek baru, sedangkan git clone digunakan untuk melanjutkan proyek yang sudah ada.

### **3. Pengertian Branch dan Manfaatnya dalam Pengembangan Fitur Baru**

Branch dalam Git bisa diibaratkan seperti jalur pengembangan yang terpisah dari kode utama. Dengan menggunakan branch, pengembang dapat bekerja pada fitur baru tanpa perlu mengubah kode di branch utama . Misalnya, ketika ingin menambahkan fitur baru seperti “form login”, kamu bisa membuat branch bernama feature-login dan melakukan semua perubahan di situ. Jika sudah selesai dan diuji, hasilnya bisa digabungkan kembali ke branch utama dengan proses merge.

Konsep ini sangat membantu terutama dalam proyek tim, karena setiap anggota bisa mengerjakan tugasnya masing-masing tanpa menimbulkan konflik dengan pekerjaan orang lain. Selain itu, branch juga membuat pengembangan menjadi lebih aman, rapi, dan fleksibel — kesalahan yang terjadi di satu branch tidak akan memengaruhi kode utama sampai perubahan tersebut benar-benar siap.

### **4. Cara Kerja GitHub dalam Mendukung Kolaborasi antar-Developer**

GitHub berperan sebagai platform berbasis web untuk menyimpan dan mengelola repository Git secara online. Dengan GitHub, para developer dapat berkolaborasi dalam satu proyek meskipun bekerja dari tempat yang berbeda. Setiap perubahan yang dilakukan di komputer lokal dapat dikirim ke GitHub menggunakan perintah push, dan sebaliknya, perubahan yang ada di repository online dapat diambil menggunakan pull.

Selain fungsi dasarnya, GitHub juga menyediakan fitur seperti pull request untuk mengusulkan perubahan sebelum digabungkan ke branch utama, issues untuk melacak bug atau ide pengembangan, serta collaboration tools yang membantu tim berdiskusi langsung di dalam proyek. Melalui mekanisme ini, GitHub menjadi jembatan utama bagi pengembang untuk bekerja secara efisien dan terorganisir dalam satu lingkungan kerja yang sama.

## 5. Alur Kerja (Workflow) Sederhana Git dari Inisialisasi hingga Push ke GitHub

Git memiliki alur kerja yang sistematis agar proses pengembangan proyek bisa lebih teratur. Workflow ini biasanya dimulai dari pembuatan repository, kemudian dilanjutkan dengan menambahkan file, melakukan commit, hingga akhirnya mengirim hasil kerja ke GitHub. Dengan mengikuti langkah-langkah ini, seluruh aktivitas pengembangan dapat terdokumentasi dengan baik dan mudah dilacak.

Tahapan umum dalam workflow Git yaitu:

- **Inisialisasi repository**

menggunakan *git init* untuk membuat repository baru, atau *git clone* jika ingin menyalin repository yang sudah ada.

- **Menambahkan file ke staging area**

menggunakan *git add <nama\_file>* agar perubahan siap disimpan.

- **Menyimpan perubahan**

ke repository lokal menggunakan *git commit -m "pesan commit"*.

- **Menghubungkan repository lokal ke GitHub**

dengan perintah *git remote add origin <url\_repository>*.

- **Mengunggah hasil kerja ke repository online**

menggunakan *git push origin main*.

## DAFTAR PUSTAKA

- Mukharom, M. S. (2015). Merancang Git Server dengan Pendekatan GitHub Social Coding dalam Peningkatan Pembelajaran Mahasiswa. *Universitas Nusantara Persatuan Guru Republik Indonesia (UNP Kediri)*.
- Sari, E. F., & Ekohariadi, E. (2021). Penerapan Github Sebagai Media E-Learning Untuk Mengetahui Keefektifan Kolaborasi Project Pada Mata Pelajaran Pemrograman Web Dan Perangkat Bergerak Di Smk Negeri 2 Surabaya. *IT-Edu: Jurnal Information Technology and Education*, 6(3), 14-22.
- Sutarman, M. D., Fadli, A., & Aliim, M. S. (2023). Perancangan Infrastruktur Devops Sistem Informasi Sentra Hki Lppm Unsoed. *TESLA: Jurnal Teknik Elektro*, 25(1), 59-71.