



PENGEMBANGAN DAN PENYELENGGARAAN  
PEMBELAJARAN DIGITAL (P3D)



# Modul Pembelajaran **SISTEM INFORMASI**

Pemrograman Web Framework



Disusun oleh : Wicaksono Yuli Sulistyio



# Modul Ajar: Dasar-Dasar Git dan GitHub

## 1. Pendahuluan

### 1.1. Tujuan Pembelajaran

Setelah menyelesaikan modul ini, peserta didik diharapkan dapat:

- Memahami konsep dasar Version Control System (VCS) dan pentingnya dalam pengembangan perangkat lunak.
- Memahami dan menggunakan Git sebagai alat VCS untuk mengelola perubahan pada proyek.
- Memahami peran GitHub sebagai layanan cloud untuk menyimpan dan berkolaborasi dalam pengembangan proyek menggunakan Git.
- Mengetahui dan memahami istilah-istilah dasar dalam Git dan GitHub.

### 1.2. Sasaran Pembelajaran

Modul ini ditujukan untuk peserta didik yang baru memulai belajar tentang:

- **Version Control System (VCS)**
- **Penggunaan Git**
- **Penggunaan GitHub untuk kolaborasi proyek**

### 1.3. Prasyarat

Peserta didik sebaiknya memiliki pemahaman dasar tentang:

- Penggunaan komputer dan internet
- Konsep dasar pemrograman (walaupun tidak wajib)

## 2. Materi Pembelajaran

### 2.1. Pengenalan Version Control System (VCS)

#### 2.1.1. Apa itu Version Control System (VCS)?

Version Control System (VCS) adalah sebuah sistem yang digunakan untuk mencatat perubahan yang dilakukan pada dokumen, program, atau informasi lainnya. VCS memungkinkan pengguna untuk mengelola dan melacak perubahan pada file, sehingga mereka dapat kembali ke versi sebelumnya jika diperlukan. VCS juga sangat penting untuk kolaborasi dalam pengembangan



perangkat lunak, karena memungkinkan banyak orang bekerja pada proyek yang sama tanpa khawatir kehilangan pekerjaan atau menimpa pekerjaan orang lain.

### 2.1.2. Manfaat VCS

- **Penyimpanan Rekaman Perubahan:** VCS menyimpan setiap perubahan yang terjadi pada proyek, memungkinkan pengguna untuk melihat sejarah perubahan.
- **Kolaborasi yang Lebih Baik:** Beberapa pengguna dapat bekerja pada proyek yang sama secara bersamaan tanpa saling mengganggu.
- **Pemulihan Perubahan:** VCS memungkinkan pengguna untuk kembali ke keadaan sebelumnya jika terjadi kesalahan dalam pengembangan.

## 2.2. Pengenalan Git

### 2.2.1. Apa itu Git?

Git adalah salah satu software VCS yang paling populer dan digunakan secara luas dalam pengembangan perangkat lunak. Git adalah sistem kontrol versi terdistribusi yang memungkinkan setiap pengguna memiliki salinan lengkap dari seluruh riwayat proyek di komputer mereka. Dengan Git, pengguna dapat membuat perubahan, mencatatnya sebagai komit, dan kemudian membagikannya dengan tim.

### 2.2.2. Istilah Dasar dalam Git

- **Repo (Repository):** Direktori atau folder proyek di mana Git mengelola semua perubahan.
- **Commit:** Sebuah rekaman/snapshot dari proyek pada saat tertentu, mencakup perubahan yang telah dilakukan.
- **Branch:** Cabang bebas dari sebuah commit, memungkinkan pengembangan fitur baru tanpa mengganggu kode utama.
- **Merge:** Proses menggabungkan dua atau lebih branch menjadi satu.
- **Remote:** Sumber yang memiliki repo yang di-host secara online, seperti di GitHub.
- **Clone:** Mengambil seluruh repo dari remote ke komputer lokal.
- **Push:** Mengirimkan commit dari repo lokal ke repo remote.
- **Pull:** Mengambil commit terbaru dari repo remote ke repo lokal.

## 2.3. Pengenalan GitHub

### 2.3.1. Apa itu GitHub?

GitHub adalah layanan cloud yang menyediakan tempat untuk menyimpan, mengelola, dan berkolaborasi pada proyek yang menggunakan Git. GitHub memungkinkan pengguna untuk menyimpan proyek mereka secara online dan bekerja sama dengan orang lain di seluruh dunia. GitHub juga menyediakan berbagai fitur seperti issue tracking, pull request, dan hosting halaman web.



### 2.3.2. Mengapa Menggunakan GitHub?

- **Kolaborasi:** GitHub memudahkan kolaborasi dengan menyediakan platform di mana pengguna dapat bekerja pada proyek yang sama, berbagi kode, dan mengelola masalah.
- **Penyimpanan Online:** Dengan GitHub, proyek dapat disimpan dengan aman di cloud, memastikan bahwa data tidak hilang.
- **Portofolio Digital:** Pengguna dapat memamerkan proyek mereka di GitHub sebagai bagian dari portofolio profesional mereka.

## 2.4. Alur Kerja Dasar dengan Git dan GitHub

### 2.4.1. Membuat Repository

- Membuat repository baru di GitHub atau di lokal komputer Anda dengan perintah `git init`.
- Menghubungkan repository lokal dengan GitHub menggunakan perintah `git remote add origin <URL>`.

### 2.4.2. Melakukan Commit dan Push

- Menambahkan file ke staging area dengan perintah `git add <file>`.
- Membuat commit dengan perintah `git commit -m "pesan commit"`.
- Mengirim commit ke GitHub dengan perintah `git push`.

### 2.4.3. Menarik Perubahan (Pull)

- Mengambil perubahan terbaru dari GitHub ke lokal menggunakan perintah `git pull`.

### 2.4.4. Bekerja dengan Branch

- Membuat branch baru untuk mengembangkan fitur tanpa mengganggu kode utama dengan perintah `git branch <nama-branch>`.
- Berpindah ke branch tersebut dengan perintah `git checkout <nama-branch>`.
- Menggabungkan branch ke branch utama setelah selesai dengan perintah `git merge <nama-branch>`.

## 3. Kesimpulan

### 3.1. Ringkasan

Dalam modul ini, peserta didik telah diperkenalkan dengan dasar-dasar penggunaan Git dan GitHub sebagai alat untuk mengelola versi proyek dan berkolaborasi dalam pengembangan perangkat lunak. Peserta diharapkan memahami pentingnya VCS dalam proyek pengembangan



perangkat lunak dan bagaimana Git serta GitHub dapat digunakan untuk memfasilitasi proses tersebut.

### 3.2. Referensi

- Dokumentasi resmi Git: <https://git-scm.com/doc>
- Panduan penggunaan GitHub: <https://guides.github.com/>