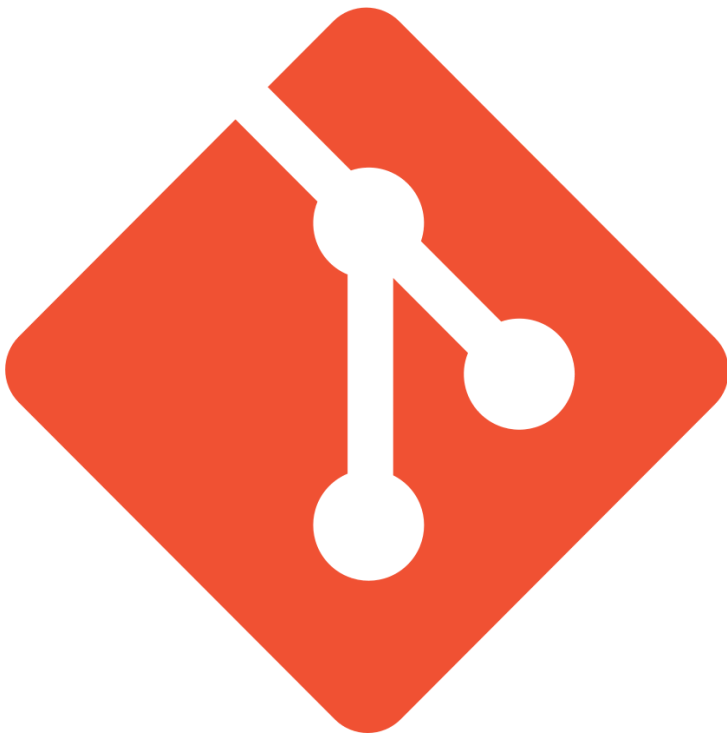


# Introduction to Git



**git**

# Git

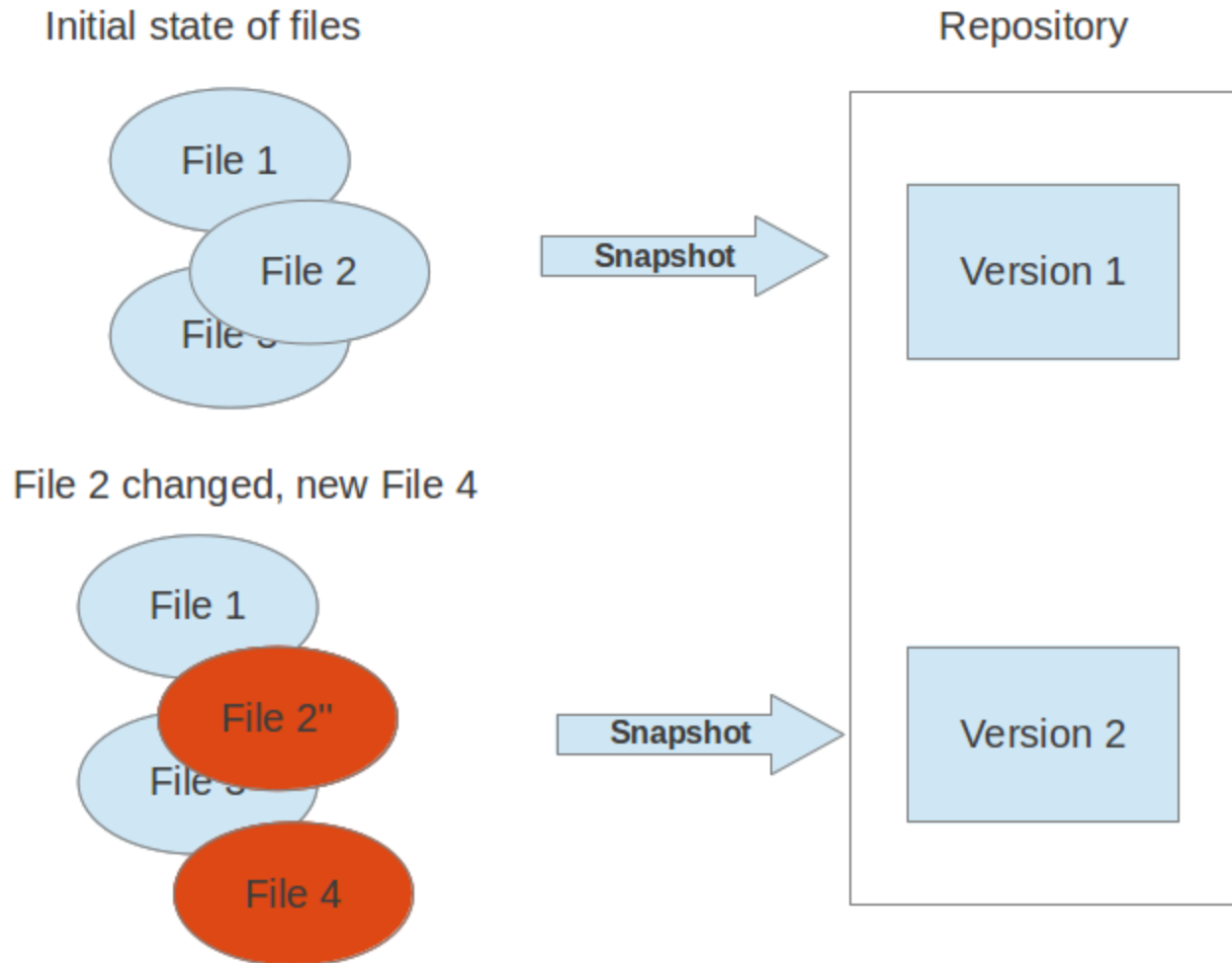
## Definisi

Git adalah salah satu Version Control System (VCS), yaitu tools untuk code versioning dan manage project coding dari kecil sampai besar.

## Versioning?

- Tracking perubahan dari kode
- Mendokumentasikan perubahan-perubahan tersebut

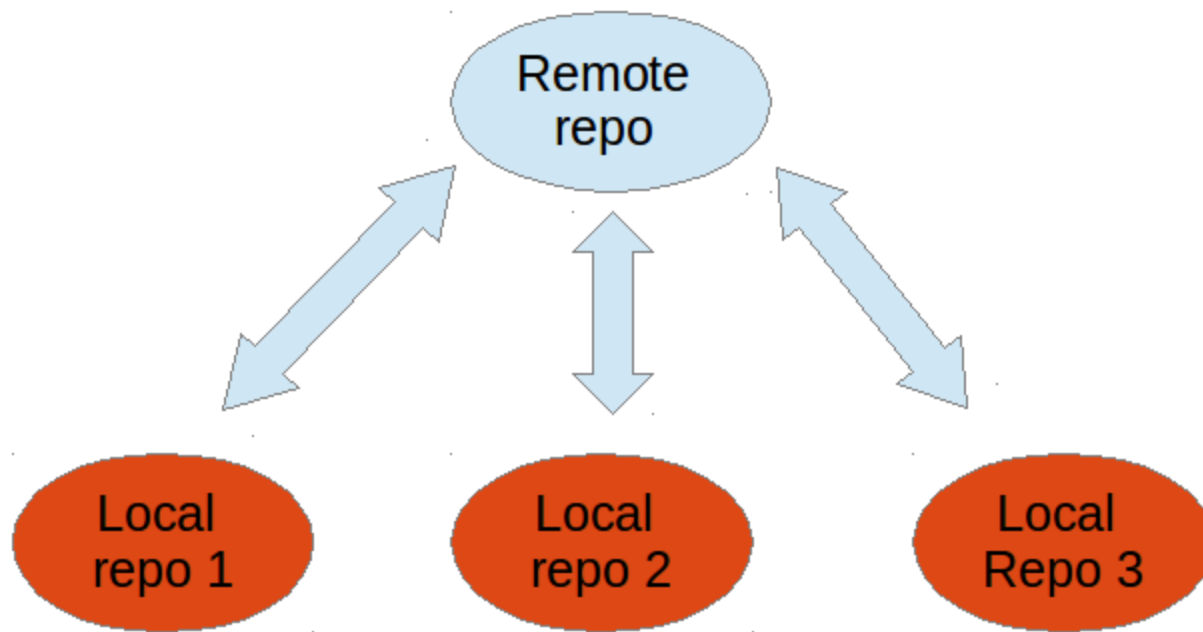
# Ilustrasi Version Control



# Jenis-Jenis Version Control System

- Localized VCS  
Menyimpan local copies dari files.
- Centralized VCS  
Menyediakan server untuk menyimpan semua files dan versionnya secara terpusat
- Distributed VCS  
Semua user memiliki copy dari repository, dan merupakan full copy (clone) dari repository tersebut

# Ilustrasi Distributed VCS



# Repository Git

- Repository, atau biasa juga disebut "Project" berisi history dari kumpulan files dimulai dari direktori tertentu.
- Proses copy repository ke local folder --> cloning. Setelah cloning, maka user akan mendapatkan repository lengkap dengan history-nya.
- Jika kita clone repository, git akan secara default berasumsi bahwa kita akan bekerja pada repository ini.

# Konsep Utama

Ada beberapa area, yaitu:

1. Workspace --> di mana user mengerjakan sesuatu (coding, dll)
2. Staging Area --> di mana user ingin menyimpan file yang ingin dimasukkan.
3. Repository --> di mana user menyimpan history dan branch (local)
4. Remote --> repository yang disimpan terpusat

# Git Interface

1. GUI
2. Command Line



# Langkah-Langkah

1. Pastikan git sudah terinstall. Berikut [tutorial install git](#).
2. Inisiasi project atau repository dengan memasukkan command

```
git init
```

atau bisa juga dengan clone project git yang sudah ada, dengan command

```
git clone url-lokasi-project-berada
```

Pastikan Anda berada di direktori yang tepat di mana Anda ingin mengerjakan project Anda

# Langkah-Langkah

2. Buat atau edit sebuah code atau text file. Misalnya, file "hello.txt" yang berisi teks "Hello World!" di dalamnya.
3. Pindahkan ke staging area, dengan command:

```
git add nama-file
```

pada kasus ini, nama-file adalah "hello.txt"

tips: pastikan selalu cek status repository, dengan command "git status"

# Langkah-Langkah

4. Commit ke repository local, dengan command

```
git commit -m "custom message"
```

Pastikan "custom message" di atas diisi oleh message yang dimengerti oleh Anda dan tim project Anda, seperti:

1. "Menambahkan file normalize.css"
2. "Menambahkan tombol forgot password di file login.php"

# Langkah-Langkah

5. Jika ingin menambahkan commit-an dan semua code ke remote repository, push semua commit-an dengan command:

```
git push origin nama_branch
```

Jika Anda tidak bisa push, berarti ada perubahan yang harus dicopy ke local repository kita terlebih dahulu. Maka, kita lakukan ini dengan command:

```
git pull lokasi_remote
```

atau, cara yang lebih aman

```
git fetch lokasi_remote
```

# Pull vs Fetch

- Pull : menggabungkan (merging) semua perubahan (file juga commit) pada remote repository ke local repository
- Fetch : import hanya commitnya saja. Command ini berguna jika kita ingin me-review terlebih dahulu perubahan sebelum menggabungkan file ke local repository.

Pull juga bisa dilaksanakan dengan mengetikkan dua command berikut:

```
git fetch lokasi_remote  
git merge nama_branch
```

# Branching

- Bayangkan sebuah tim project yang besar (dari mulai developer hingga tester), ingin menambah banyak fitur baru untuk sebuah produk.
- Bagaimana tim tersebut bisa mengintegrasikan fitur-fitur baru tersebut sambil melakukan testing terhadap fitur baru tersebut tanpa membuat code berantakan?
- Jawaban: Branches, di mana tim bisa bekerja pada "direktori baru", yang nantinya bisa dihapus atau di merge.

# Branching

- Bayangkan branch adalah folder-folder yang terpisah, yang masing-masing berisi file beserta history.



# Branching

- Command untuk menggabungkan file dan history antar branch

```
git merge nama_branch
```



# Good Git Practice

1. Sering sering check in (add dan commit), terutama ketika code kita sudah bekerja atau memenuhi spek yang diberikan.
2. Tulis commit message dengan detail dengan kalimat yang bisa dimengerti oleh semua orang.
3. Untuk fitur-fitur baru, lakukan branching.