



Pertemuan 4 : Version Control System

Bahasa Pemrograman

Agung Nugroho, M.Kom

Teknik Informatika – S1

Fakultas Teknik

Universitas Pelita Bangsa



Agung Nugroho, S.Kom, M.Kom

- 1994 | SDN Pulau Panggung, OKU Sumsel
- 1997 | MTs Lab Fak Tarbiah IAIN SUKA, Yogyakarta
- 2000 | SMK PIRI 1, Yogyakarta
- 2004 | Ilmu Komputer, Universitas Ahmad Dahlan, Yogyakarta
- 2016 | Magister Komputer, STMIK Eresha, Jakarta

- 2012 - Present | Freelance Web Developer
- 2011 - 2012 | Web Developer at BP Indonesia
- 2010 - 2011 | OSS Core Engineer at PT Ericsson Indonesia
- 2008 - 2009 | Radio Database Planner at PT. NextWave subcon NSN
- 2005 - 2008 | Software Developer at PT Gamatechno Indonesia
- 2004 - 2005 | Web Programmer at PT Reftindo Sarana

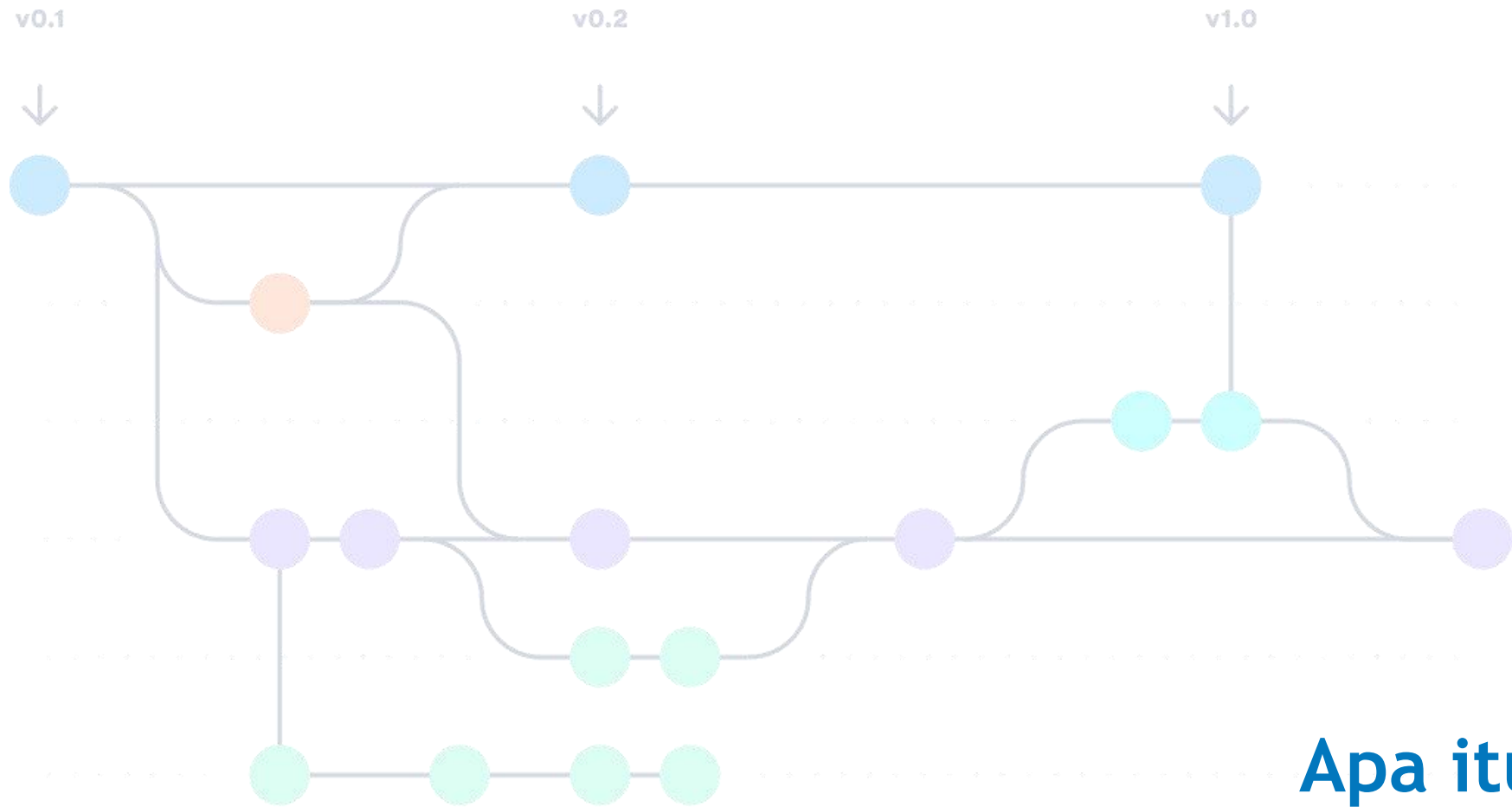


- www.linkedin.com/in/kangmasagung
- www.fb.me/agung.n
- www.koding.web.id



Mengenal Version Control System

Pertemuan 4



Apa itu

Version Control System?



MASTER



HOTFIX



RELEASE



DEVELOP

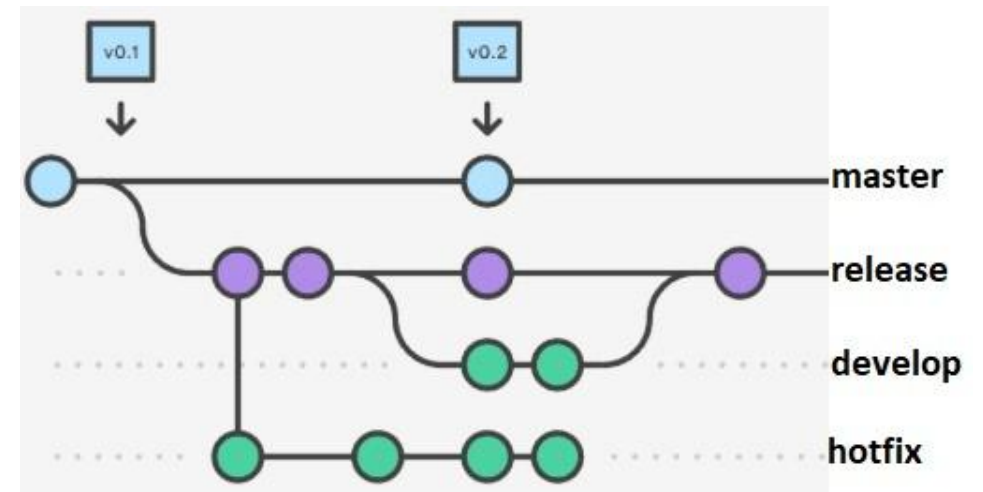


FEATURES



Apa itu VCS

- VCS atau *Version Control System* merupakan sebuah sistem yang merekam perubahan-perubahan dari sebuah berkas atau sekumpulan berkas dari waktu ke waktu sehingga Anda dapat melihat kembali setiap perubahannya.
- Salah satu DVCS (*Distributed Version Control System*) yang sangat populer saat ini adalah git.





Apa itu Git?

- Git adalah salah satu sistem pengontrol versi (Version Control System) pada proyek perangkat lunak yang diciptakan oleh Linus Torvalds.
- Pengontrol versi bertugas mencatat setiap perubahan pada file proyek yang dikerjakan oleh banyak orang maupun sendiri.
- Git dikenal juga dengan distributed revision control (VCS terdistribusi), artinya penyimpanan database Git tidak hanya berada dalam satu tempat saja.



Instalasi Git

- Download **Git**, buka website resminya Git (git-scm.com).
- Kemudian unduh Git sesuai dengan arsitektur komputer kita. Kalau menggunakan 64bit, unduh yang 64bit. Begitu juga kalau menggunakan 32bit.
- Selamat, Git sudah terinstal di Windows. Untuk mencobanya, silahkan buka **CMD** atau **PowerShell**, kemudian ketik perintah

git --version.

```
C:\> Command Prompt
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\Development>git --version
git version 2.11.1.windows.1

C:\Users\Development>
```



Menambahkan Global Config

- Pada saat pertama kali menggunakan git, perlu dilakukan konfigurasi *user.name* dan *user.email*
- konfigurasi ini bisa dilakukan untuk global repository atau individual repository.
- apabila belum dilakukan konfigurasi, akan mengakibatkan terjadi kegagalan saat menjalankan perintah `git commit`
- Config Global Repository

```
$ git config --global user.name "nama_user"
```

```
$ git config --global user.email "nama_user"
```




Perintah Dasar Git

- **git init**, perintah untuk membuat repository local
- **git add**, perintah untuk menambahkan file baru, atau perubahan pada file pada staging sebelum proses commit.
- **git commit**, perintah untuk menyimpan perubahan kedalam database git.
- **git push -u origin master**, perintah untuk mengirim perubahan pada repository local menuju server repository.
- **git clone [url]**, perintah untuk membuat working directory yang diambil dari repository server.
- **git remote add origin [url]**, perintah untuk menambahkan remote server/repository server pada local repository (*working directory*)
- **git pull**, perintah untuk mengambil/mendownload perubahan terbaru dari server repository ke local repository



Membuat Reposiory Local

- Buka direktory aktif, misal: **d:\labs_pemrograman1** (buka menggunakan Windows Explorer)
- klik kanan pada direktory aktif tersebut, dan pilih menu **Git Bash**, sehingga muncul *git bash commad*
- Buat direktory project praktikum pertama dengan nama **latihan1**

```
$ mkdir latihan1  
$ cd latihan1
```

- Sehingga terbentuk satu direktori baru dibawahnya, selanjutnya masuk kedalam direktori tersebut dengan perintah **cd** (*change directory*)
- direktory aktif menjadi: **d:\labs_pemrograman1\latihan1**



Membuat Reposiory Local

- Jalankan perintah **git init**, untuk membuat repository local.

```
$ git init
```

- Repository baru berhasil di inisialisasi, dengan terbentuknya satu direktori hidden dengan nama **.git**
- Pada direktori tersebut, semua perubahan pada *working directory* akan disimpan.



Menambahkan File baru pada repository

- Untuk membuat file dapat menggunakan text editor, lalu menyimpan filenya pada direktori aktif (repository)
- disini kita akan coba buat satu file bernama README.md (text file)

```
$ echo "# Latihan 1" >> README.md
```

- File **README.md** berhasil dibuat.

```
chung@thinkpad-x201: ~/labpy/latihan1
chung@thinkpad-x201:~/labpy/latihan1$ echo "#Latihan 1" >> README.md
chung@thinkpad-x201:~/labpy/latihan1$ ls -l
total 4
-rw-rw-r-- 1 chung chung 22 Jan 20 15:29 README.md
chung@thinkpad-x201:~/labpy/latihan1$
```



Menambahkan File baru pada repository

- Untuk menambahkan file yang baru saja dibuat tersebut gunakan perintah **git add**.

```
$ git add README.md
```

- File **README.md** berhasil ditambahkan.

```
total 4
-rw-rw-r-- 1 chung chung 33 Jan 20 20:06 README.md
chung@thinkpad-x201:~/labpy/latihan1$ git add README.md
chung@thinkpad-x201:~/labpy/latihan1$ git status
On branch master

Initial commit

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)

    new file:   README.md

chung@thinkpad-x201:~/labpy/latihan1$
```



Commit (Menyimpan perubahan ke database)

- Untuk menyimpan perubahan yang ada kedalam database repository local, gunakan perintah **git commit -m “komentar commit”**

```
$ git commit -m "File pertama saya"
```

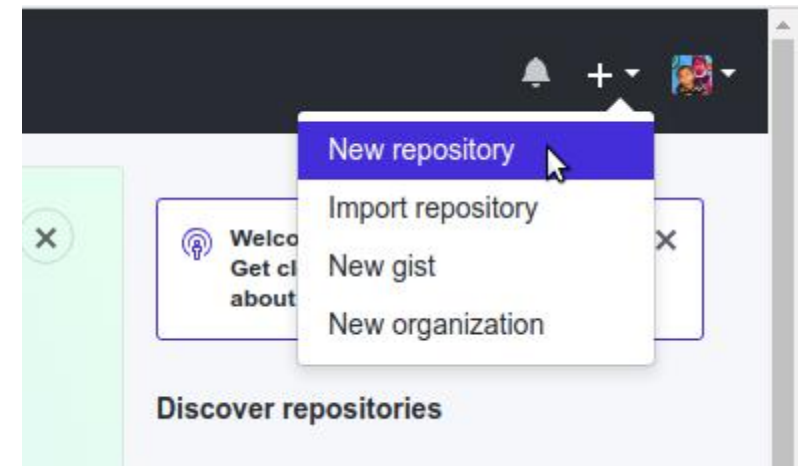
- Perubahan berhasil disimpan.

```
chung@thinkpad-x201:~/labpy/latihan1$ git commit -m "File pertama saya"
[master (root-commit) 2ce0996] File pertama saya
1 file changed, 3 insertions(+)
create mode 100644 README.md
chung@thinkpad-x201:~/labpy/latihan1$
```



Membuat repository server

- Server repository yang akan kita gunakan adalah <http://github.com>
- Anda harus membuat akun terlebih dahulu.
- Pada laman github, klik tombol start a project, atau
- Dari menu (icon +) klik New Repository





Membuat repository server



- Isi nama repositorynya, misal: **labpy1**.
- lalu klik tombol **Create repository**

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

Repository name *

 abuazzam ▾ / labpy 

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

Description (optional)

☒  **Public**

Anyone can see this repository. You choose who can commit.

☐  **Private**

You choose who can see and commit to this repository.

☐ **Initialize this repository with a README**

This will let you immediately clone the repository to your computer. Skip this step if you're imp

Add .gitignore: **None** ▾

Add a license: **None** ▾



Create repository



Menambahkan Remote Repository

- Remote Repository merupakan repository server yang akan digunakan untuk menyimpan setiap perubahan pada local repository, sehingga dapat diakses oleh banyak user.
- Untuk menambahkan remote repository server, gunakan perintah **git remote add origin [url]**

```
$ git remote add origin https://github.com/abuazzam/labpy1.git
```



Push (Mengirim perubahan ke server)

- Untuk mengirim perubahan pada local repository ke server gunakan perintah git push.

```
$ git push -u origin master
```

- Perintah ini akan meminta memasukkan username dan password pada akun github.com

```
1 file changed, 1 insertion(+)
chung@thinkpad-x201:~/labpy/latihan1$ git push -u origin master
Username for 'https://github.com': abuazzam
Password for 'https://abuazzam@github.com':
Counting objects: 7, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (7/7), 714 bytes | 0 bytes/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To https://github.com/abuazzam/labpy1.git
    03a4aab..9d829e4  master -> master
Branch master set up to track remote branch master from origin.
chung@thinkpad-x201:~/labpy/latihan1$
```



Melihat hasilnya pada server repository

- Buka laman github.com, arahkan pada repository-nya.
- Maka perubahan akan terlihat pada laman tersebut.

The screenshot shows the GitHub interface for the repository 'abuazzam / labpy1'. At the top, there are navigation tabs: 'Code' (selected), 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', 'Insights', and 'Settings'. To the right of these tabs are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below the tabs, there is a message: 'No description, website, or topics provided.' with an 'Edit' button. Underneath, there is a 'Manage topics' link. A summary bar shows '4 commits', '1 branch', '0 releases', and '1 contributor'. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and a green 'Clone or download' button. The main content area shows a commit by 'abuazzam' with the message 'File pertama saya' and the commit hash '9d829e4' from '4 minutes ago'. Below the commit, there is a file named 'README.md' with the content '#Latihan 1 #Latihan 1'.



Clone Repository

- Clone repository, pada dasarnya adalah meng-copy repository server dan secara otomatis membuat satu direktory sesuai dengan nama repositorynya (*working directory*).
- Untuk melakukan cloning, gunakan perintah **git clone [url]**

```
chung@thinkpad-x201:~/labpy/labs$ git clone https://github.com/abuazzam/labpy1.git
Cloning into 'labpy1'...
remote: Enumerating objects: 10, done.
remote: Counting objects: 100% (10/10), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 10 (delta 1), reused 9 (delta 0), pack-reused 0
Unpacking objects: 100% (10/10), done.
Checking connectivity... done.
chung@thinkpad-x201:~/labpy/labs$ ls -l
total 4
drwxrwxr-x 3 chung chung 4096 Jan 20 20:42 labpy1
chung@thinkpad-x201:~/labpy/labs$
```



Kegunaan file README.md

- Apabila kita menggunakan github, untuk memberikan penjelasan awal pada project yang kita buat, maka dapat menggunakan sebuah file yang bernama README.md
- Pada file tersebut kita dapat membuat dokumentasi awal dari setiap project yang kita buat untuk memberikan penjelasan atau sekedar cara penggunaan dari aplikasi yang kita kembangkan.
- Penulisan file README.md berbasis teks, dan untuk pemformatannya menggunakan **Markdown** format.
- untuk lebih jelasnya, dapat anda pelajari cara penggunaan markdown pada url berikut: <https://guides.github.com/features/mastering-markdown/>



Any Question?



Latihan

- Buatlah satu repository **LatihanVCS**
- Buat file **README.md**, lalu isi file tersebut dengan penjelasan (**tutorial**) cara penggunaan **git**, dan langkah-langkahnya lengkapi juga dengan ***screenshot*** prosesnya.



Terimakasih

Agung Nugroho

agung@pelitabangsa.ac.id

www.koding.web.id