

# Git Fundamentals

Presented by Fei

# Sub Topics

- ◆ Git & GitHub
- ◆ Repository Area
- ◆ Git Commands
- ◆ Git Init, Add, and Commit
- ◆ Git Branch & Merge
- ◆ GitHub Branch & Merge
- ◆ Fork
- ◆ Remote and Multiple Remote
- ◆ GitHub Pages
- ◆ .gitignore
- ◆ Git & Web Hosting



# Git & GitHub

Git merupakan salah satu Version Control System (VCS) dalam bentuk software yang digunakan untuk mengelola repository yang terdapat pada suatu folder yang kita miliki dengan tujuan untuk melakukan tracking pada perubahan yang terjadi pada folder tersebut.

Keuntungan menggunakan VCS:  
versioning, collaboration, branching, merging, dan recovery

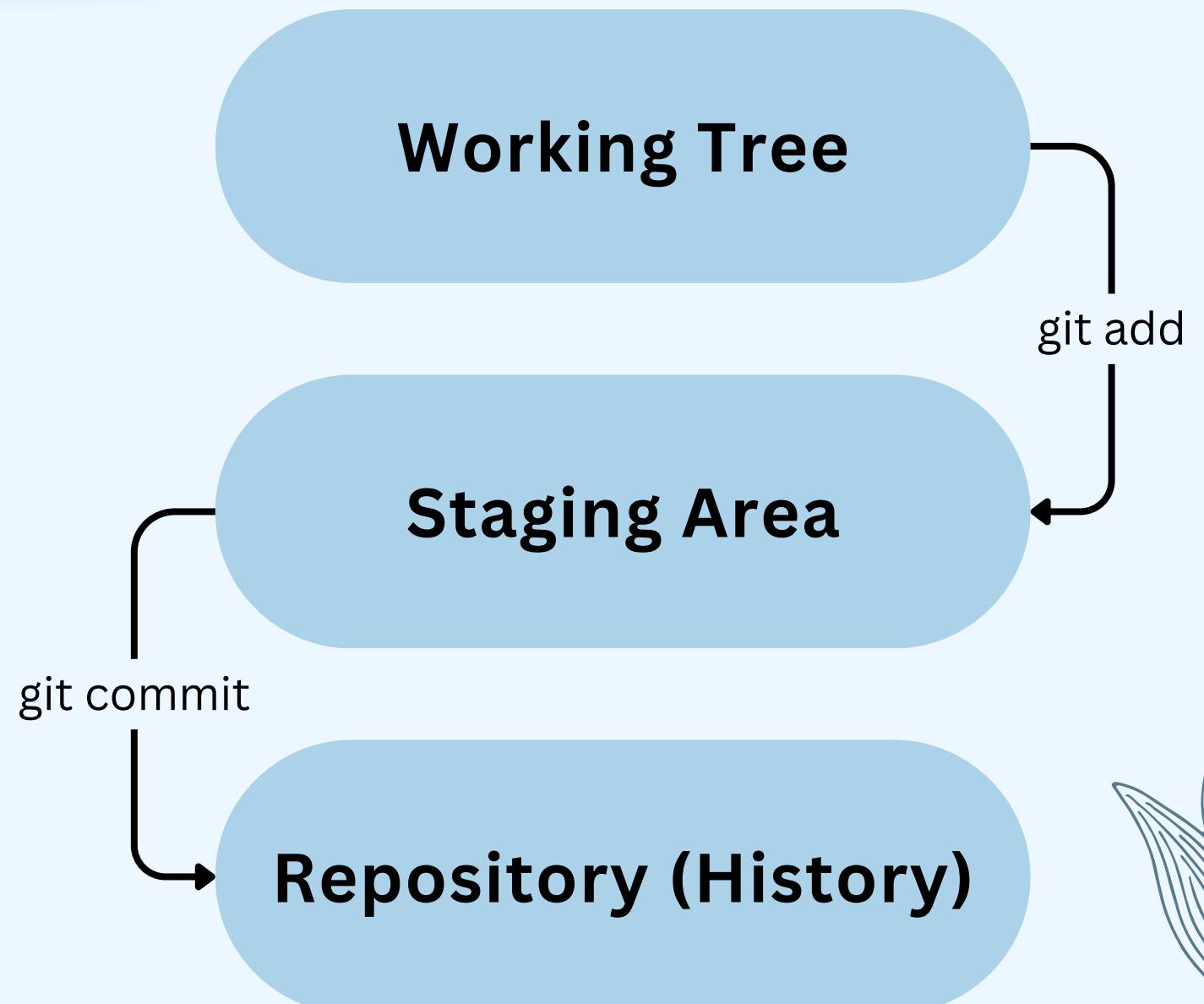
GitHub merupakan sebuah layanan cloud yang memberikan usernya kebebasan untuk menyimpan dan mengatur repository yang mereka miliki secara online dan berkolaborasi dengan menggunakan Git.

# Repository Area

Repository adalah folder yang terdapat dalam local kita dan digunakan sebagai working directory yang akan dipantau oleh git.

Area (3):

- Working tree: folder tempat kita bekerja
- Staging area: area yang menyimpan perubahan yang terjadi dalam working tree (add)
- History: area yang menyimpan history commit yang berasal dari staging area dan sudah disetujui (commit -m)



# GitHub Repository

GitHub memberikan interface yang dapat mempermudah kita dalam membuat dan mengakses repository yang kita miliki namun masih memiliki ketergantungan dengan Git untuk mengatur local repository

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (\*).

Owner \* fangelineap / Repository name \* yapping-time yapping-time is available.

Great repository names are short and memorable. Need inspiration? How about [bug-free-engine](#)?

Description (optional) Repo untuk menyimpan keluh kesah

Public Anyone on the internet can see this repository. You choose who can commit.

Private You choose who can see and commit to this repository.

Initialize this repository with:  Add a README file This is where you can write a long description for your project. [Learn more about READMEs](#).

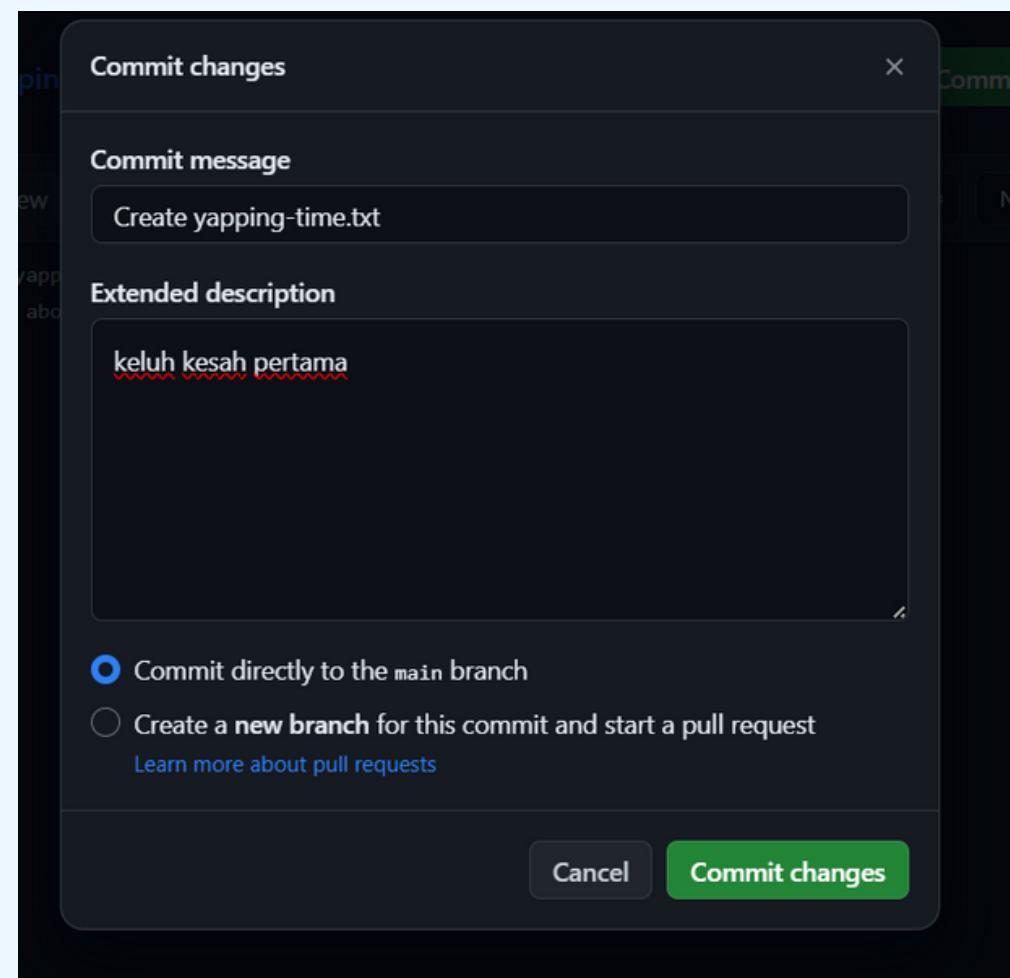
Add .gitignore .gitignore template: None Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license License: None A license tells others what they can and can't do with your code. [Learn more about licenses](#).

This will set `main` as the default branch. Change the default name in your [settings](#).

You are creating a public repository in your personal account.

[Create repository](#)



Commit

Remove the first line i don't like yapping

main fangelineap committed 3 minutes ago

1 parent 1500a8e commit 6e1a574

Showing 1 changed file with 1 addition and 2 deletions.

Whitespace Ignore whitespace Split Unified

3 - I love yapping  
2 - I think about yapping all the time  
1 + I think about yapping all the time  
3 - But I don't have the neergy to yap  
2 + But I don't have the energy to yap

Prompt: Advantages of using GitHub

# Git Commands

- Git init : menginisialisasi sebuah repository pada sebuah local folder
- Git add <file(s)> : menambahkan files ke staging area sebelum melakukan commit
- Git commit : proses melakukan perubahan pada repository dengan pesan yang spesifik
- Git checkout <branchName> : berpindah pada satu branch ke branch lain
- Git clone <repo-url> : menduplikat sebuah repository dari github menjadi local repository
- Git push : mengusulkan perubahan yang telah dicommit pada local repository pada remote repository
- Git pull : fetch perubahan dari remote repository pada local repository agar up-to-date
- Git branch : memberikan list dari branch yang terdapat pada sebuah repository
- Git merge : menggabungkan perubahan dari sebuah branch pada 1 branch spesifik
- Git help : memberikan list bantuan mengenai cara menggunakan git commands
- Git log : memberikan list history commit yang pernah terjadi dalam suatu repository

**Prompt:** Git commands

# Git Init, Add, and Commit Changes

```
MINGW64:/d/Binus/Intern/Learning Materials/GitGithub/git-example
fange@LAPTOP-3AIV3NUB MINGW64 ~
$ cd d:
fange@LAPTOP-3AIV3NUB MINGW64 /d
$ cd Binus/Intern/Learning\ Materials/GitGithub/
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub
$ cd git-example
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example
$ git init
Initialized empty Git repository in D:/Binus/Intern/Learning Materials/GitGithub
/git-example/.git/
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ ls
index.html
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ git status
On branch main
No commits yet
Untracked files:
  (use "git add <file>..." to include in what will be committed)
    index.html

nothing added to commit but untracked files present (use "git add" to track)
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ git add index.html
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ git status
On branch main
No commits yet
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   index.html
```

## Git Init

Untuk menginisialisasi folder tertentu agar dapat menjadi folder yang dipantau perubahannya oleh git

## Git Add & Commit Changes

Git add untuk menambahkan semua file yang memiliki perubahan pada staging area dan commit digunakan untuk finalize perubahan

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ git commit
Aborting commit due to empty commit message.

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ git commit -m 'creating new index.html file'
[main (root-commit) a42f8b9] creating new index.html file
  1 file changed, 11 insertions(+)
   create mode 100644 index.html

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example (main)
$ |
```

# Restoring a Commit

Git log digunakan untuk melihat history dari seluruh commit yang telah terjadi pada suatu repository. Setiap commit akan terdiri dari identifier, nama dan waktu commit, serta message dari setiap commit.

Git checkout juga dapat digunakan untuk berpindah commit.

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/gitignore ((ba7f0be...))
$ git log
commit ba7f0bee632340618764d0e8ff5514f3cae1eb22 (HEAD, origin/main, main)
Author: Fangeline <fangelineap@gmail.com>
Date:   Tue Feb 20 18:07:44 2024 +0700

    modify gitignore file

commit aee4b01fb0e3a9f0860b8e7f96f094e202cd405a
Author: Fangeline <fangelineap@gmail.com>
Date:   Tue Feb 20 18:06:22 2024 +0700

    adding gitignore file

commit 83d3d2a6a94f01b12946fdb968992ef2ffcc4c9d
Author: Fangeline <fangelineap@gmail.com>
Date:   Tue Feb 20 18:00:54 2024 +0700

    add 2 files

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/gitignore ((ba7f0be...))
$ git checkout '83d3d2a'
Previous HEAD position was ba7f0be modify gitignore file
HEAD is now at 83d3d2a add 2 files
```

**Prompt:** Restoring commit in Git

# Git Branch

Branch memungkinkan programmer untuk bekerja pada bagian repository yang lain tanpa menganggu branch utama dari repo tersebut

Git bash:

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-e
xample2 (main)
$ git checkout dosen
Switched to branch 'dosen'

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-e
xample2 (dosen)
$ git branch
* dosen
  main
  staff

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-e
xample2 (dosen)
$ |
```

Branch main:

Name	Date modified	Type
.git	2/20/2024 12:55 PM	File folder
mahasiswa.html	2/20/2024 12:46 PM	Chrome HTML Do...

Branch dosen:

Name	Date modified	Type
.git	2/20/2024 12:56 PM	File folder
mahasiswa.html	2/20/2024 12:46 PM	Chrome HTML Do...
dosen.html	2/20/2024 12:56 PM	Chrome HTML Do...

Prompt: Git branch and merge

# Git Merge

Operasi untuk menggabungkan perubahan yang terdapat pada 2 branch menjadi 1.

Fast forward merge:

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (staff)
$ git checkout main
Switched to branch 'main'

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ git merge dosen
Updating ec086aa..8c56fa1
Fast-forward
  dosen.html | 14 ++++++
  1 file changed, 14 insertions(+)
  create mode 100644 dosen.html

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ git branch
  dosen
* main
  staff

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ git branch -d dosen
Deleted branch dosen (was 8c56fa1).
```

.git	2/20/2024 02:44 PM	File folder
dosen.html	2/20/2024 02:43 PM	Chrome HTML Do... 1 KB
mahasiswa.html	2/20/2024 12:46 PM	Chrome HTML Do... 1 KB

Three way merge:

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ git merge staff
Merge made by the 'ort' strategy.
  staff.html | 14 ++++++
  1 file changed, 14 insertions(+)
  create mode 100644 staff.html

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ graph
* dffaf30 (HEAD -> main) Merge branch 'staff'
|\ 
| * c919f2d (staff) menambahkan file staff
| * 8c56fa1 menambahkan file dosen
|/
* ec086aa menambahkan data satu mahasiswa
* dde5e97 menambahkan data mahasiswa

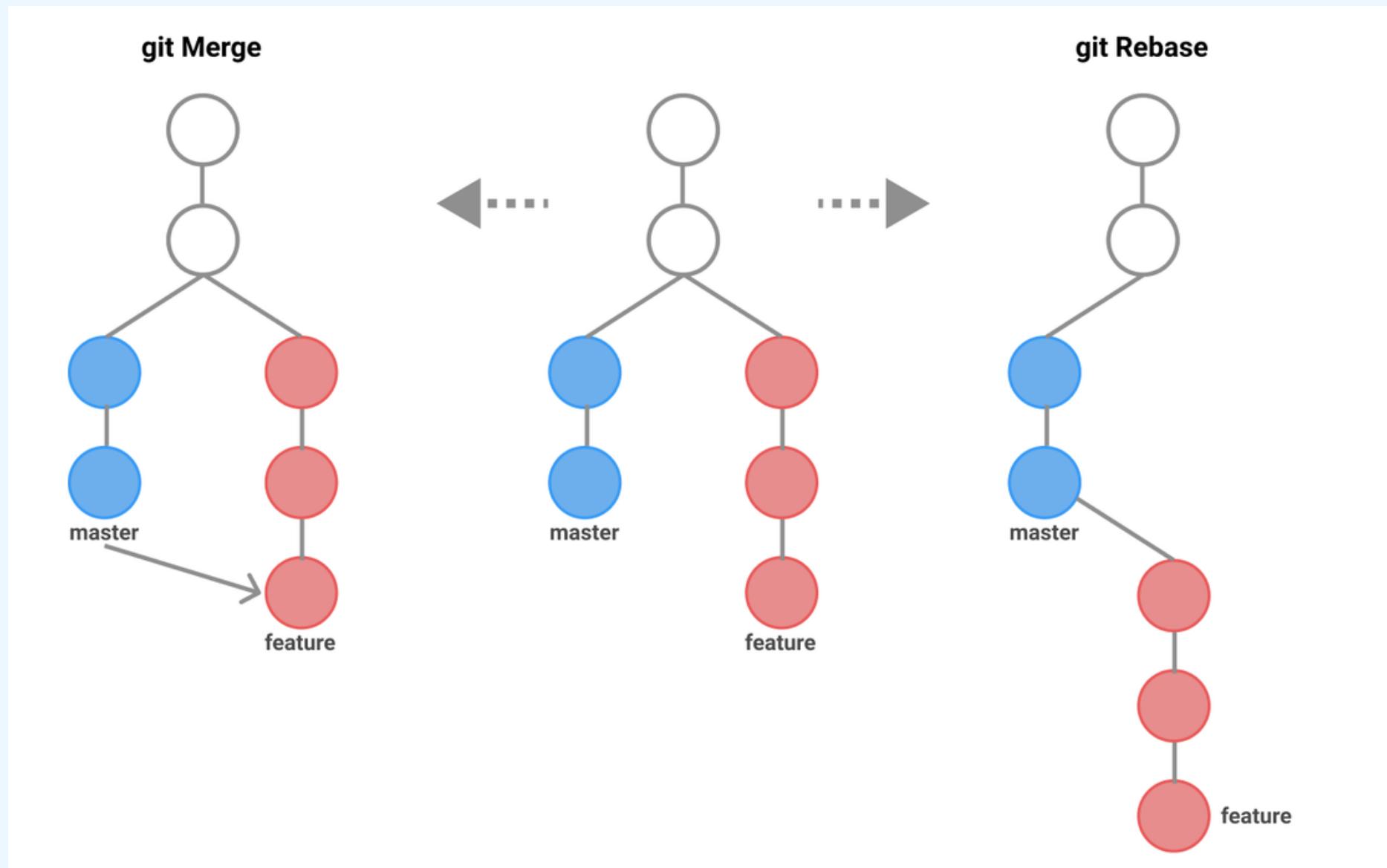
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ git branch -d staff
Deleted branch staff (was c919f2d).
```

.git	2/20/2024 02:49 PM	File folder
dosen.html	2/20/2024 02:43 PM	Chrome HTML Do... 1 KB
mahasiswa.html	2/20/2024 12:46 PM	Chrome HTML Do... 1 KB
staff.html	2/20/2024 02:49 PM	Chrome HTML Do... 1 KB

Prompt: Git branch and merge

# Rebase

Rebase memiliki tujuan yang sama dengan fast forward dan three way merge. Rebase memindahkan suatu branch agar terdapat pada akar dari branch lain sehingga dapat melakukan fast forward merge. Setelah di rebase, kita hanya tinggal menyamakan branch master dan feature menjadi commit terakhir dan push ke repo.



Prompt: Rebase merge

# Git Merge Conflict

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ graph
* fb8db42 (HEAD -> main) mengubah file mahasiswa
| * 79b51ac (dev) mengubah file mahasiswa
|/
* dffaf30 Merge branch 'staff'
|\
| * c919f2d menambahkan file staff
* | 8c56fa1 menambahkan file dosen
|/
* ec086aa menambahkan data satu mahasiswa
* dde5e97 menambahkan data mahasiswa

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ git merge dev
Auto-merging mahasiswa.html
CONFLICT (content): Merge conflict in mahasiswa.html
Automatic merge failed; fix conflicts and then commit the result.

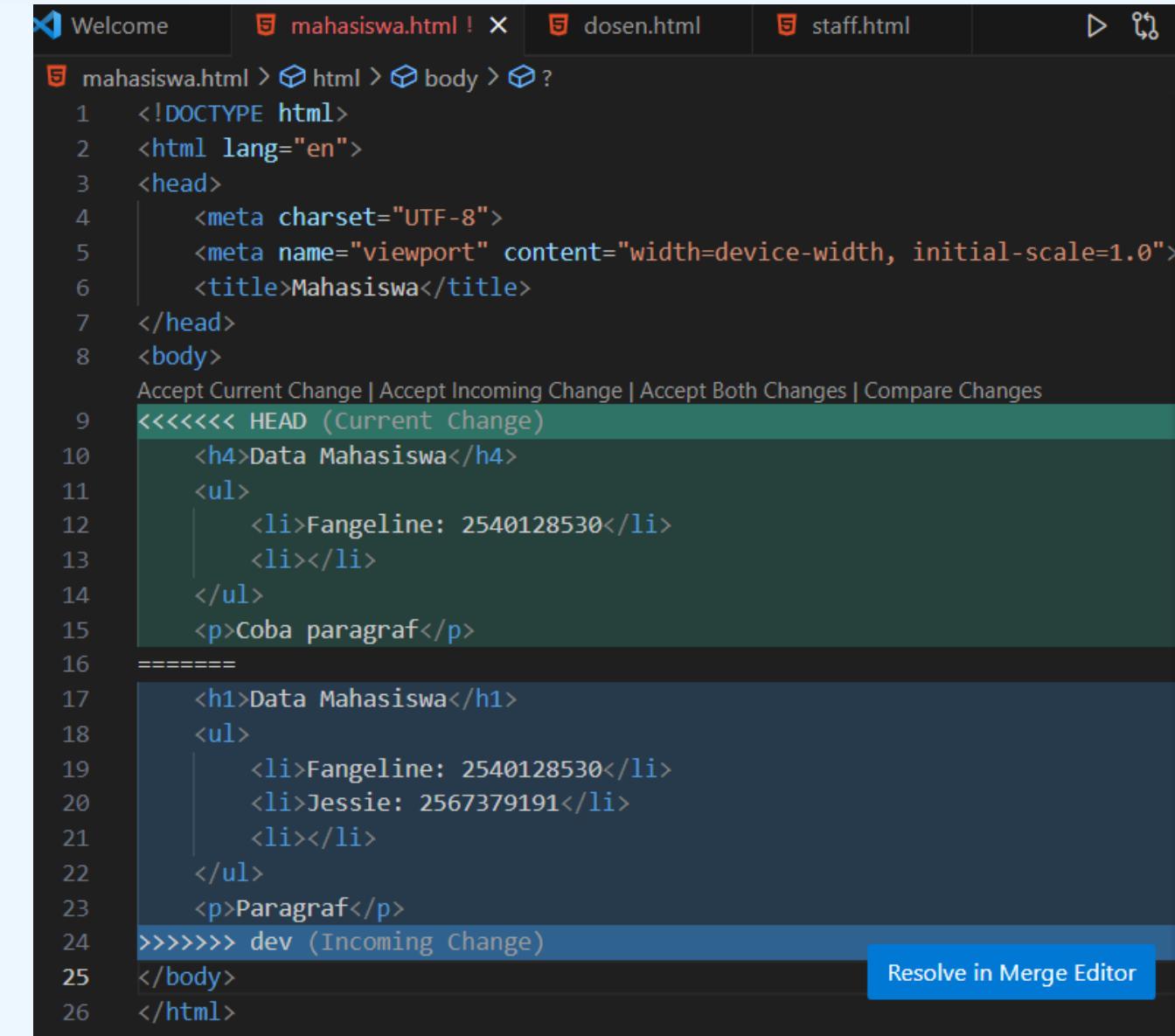
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main|MERGING)
$ git status
on branch main
You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

unmerged paths:
  (use "git add <file>..." to mark resolution)
    both modified:  mahasiswa.html

no changes added to commit (use "git add" and/or "git commit -a")

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main|MERGING)
$ git commit -am 'merge branch dev'
[main fcbd2b1] merge branch dev

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-example2 (main)
$ graph
* fcbd2b1 (HEAD -> main) merge branch dev
|\
| * 79b51ac (dev) mengubah file mahasiswa
* | fb8db42 mengubah file mahasiswa
|/
* dffaf30 Merge branch 'staff'
|\
| * c919f2d menambahkan file staff
* | 8c56fa1 menambahkan file dosen
|/
* ec086aa menambahkan data satu mahasiswa
* dde5e97 menambahkan data mahasiswa
```



The screenshot shows a code editor with the file `mahasiswa.html` open. The code is an HTML document with the following structure:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Mahasiswa</title>
</head>
<body>
  <h4>Data Mahasiswa</h4>
  <ul>
    <li>Fangeline: 2540128530</li>
    <li></li>
  </ul>
  <p>Coba paragraf</p>
  =====
  <h1>Data Mahasiswa</h1>
  <ul>
    <li>Fangeline: 2540128530</li>
    <li>Jessie: 2567379191</li>
    <li></li>
  </ul>
  <p>Paragraf</p>

```

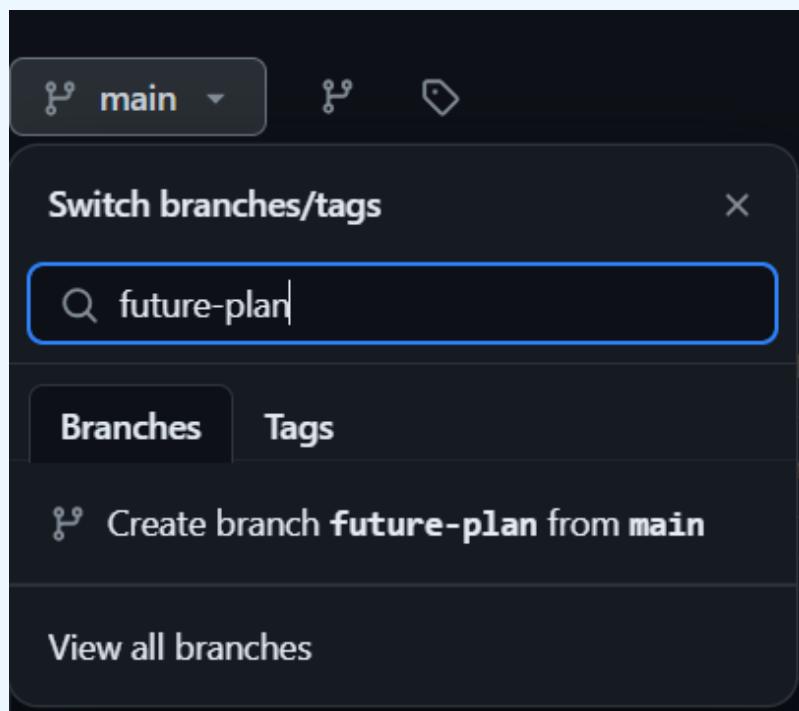
A conflict is indicated by the text `CONFLICT (content): Merge conflict in mahasiswa.html`. The code editor interface includes tabs for `mahasiswa.html`, `dosen.html`, and `staff.html`. A toolbar at the top right includes icons for file operations like `Accept Current Change`, `Accept Incoming Change`, `Accept Both Changes`, and `Compare Changes`. A blue bar at the bottom right contains the button `Resolve in Merge Editor`.

Conflict terjadi karena terdapat 2 branch yang memiliki perubahan pada line atau section dari code yang sama.

Prompt: Git branch and merge

# Github Branch

Membuat branch baru pada repository yang kita inginkan dengan tujuan agar kita dapat melakukan perubahan tanpa memberikan efek pada branch asli.



A screenshot of the GitHub 'Branches' page. At the top right, there is a green button labeled 'New branch'. The page has tabs for 'Overview', 'Yours', 'Active', 'Stale', and 'All', with 'Yours' being active. Below the tabs is a search bar with the placeholder 'Search branches...'. The main area is divided into three sections: 'Default', 'Your branches', and 'Active branches'. In each section, there is a table with columns for 'Branch', 'Updated', 'Check status', 'Behind/Ahead', and 'Pull request'. The 'Default' section shows a 'main' branch updated '1 minute ago' and marked as 'Default'. The 'Your branches' section shows a 'future-plan' branch updated '6 minutes ago' with a pull request icon. The 'Active branches' section also shows the 'future-plan' branch updated '6 minutes ago' with a pull request icon.

Prompt: GitHub branch and merge

# Github Merge

Menggabungkan perubahan yang terdapat pada 2 file berbeda menjadi 1 pada VCS virtual.

The screenshot shows a GitHub repository interface. At the top, a banner indicates that the 'future-plan' branch has recent pushes 1 minute ago. A green button labeled 'Compare & pull request' is visible. Below the banner, the repository navigation bar includes dropdowns for 'future-plan' and 'main', and buttons for 'Go to file', '+', and 'Code'. A message states 'This branch is 2 commits ahead of main.' The commit history shows three commits:

- A commit by 'fangelineap' titled 'Update yapping-time.txt' made 1 minute ago.
- An initial commit by 'fangelineap' for 'README.md' made 30 minutes ago.
- A second commit by 'fangelineap' titled 'Update yapping-time.txt' made 1 minute ago.

Pull request (PR) dilakukan agar perubahan yang kita buat pada repository kita dapat dimerge pada repository asli (biasanya) milik orang lain. PR yang kita lakukan akan berisikan snapshot dari perubahan code yang dilakukan sehingga pemilik repo dapat mereview code sebelum merge.

**Prompt:** GitHub branch and merge

The first screenshot shows the 'Open a pull request' page with 'base: main' and 'compare: future-plan'. It indicates that the branches can be automatically merged. The second screenshot shows a merged pull request with a title 'Future plan' and a detailed description. The third screenshot shows the conversation tab of the pull request, where 'fangelineap' has commented to add more context to the merge.

# Github Merge Conflict

Terjadi saat line atau section dari code yang sama mengalami suatu perubahan dari 2 atau lebih branch.

⌚ **activity-1** had recent pushes 1 minute ago Compare & pull request

⌚ **activity-2** had recent pushes 39 seconds ago Compare & pull request

---

 main   [Go to file](#)   [Code](#) ▾

---

File	Commit Message	Time Ago
 README.md	Initial commit	41 minutes ago
 yapping-time.txt	Adding one thing	3 minutes ago

# Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks. [Learn more about diff comparisons here.](#)

base: main ▾ ← ... compare: activity-2 ▾

**Can't automatically merge.** Don't worry, you can still create the pull request.

**Add a title**

Update yapping-time.txt

**Add a description**

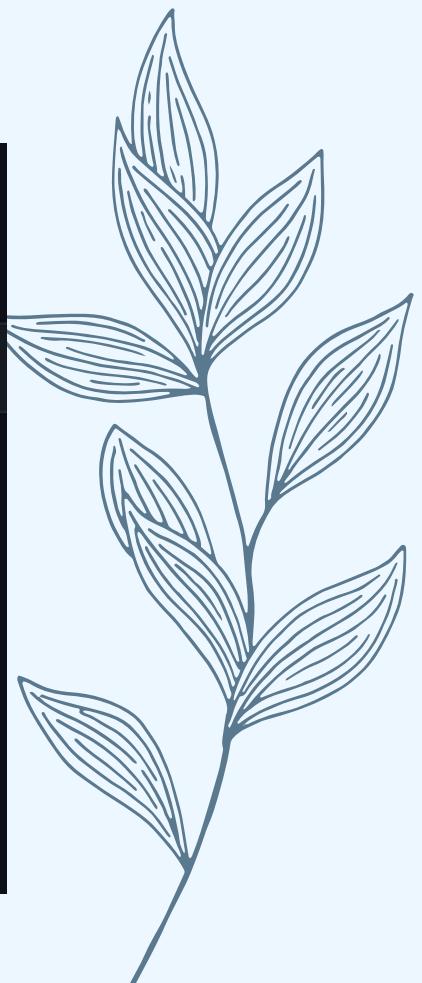
Write Preview

Add your description here...

# Update yapping-time.txt #3

Resolving conflicts between `activity-2` and `main` and committing changes → `activity-2`

1 conflicting file	yapping-time.txt	1 conflict	Prev ^	Next v	⚙️ ▾	Mark as resolved
 yapping-ti... yapping-time.txt	<pre>1 I think about yapping all the time 2 But I don't have the energy to yap 3 Yapping is tiring 4 So I use my energy to knit most of the time 5 And crochet too 6 I made a new sweater last week 7 &lt;&lt;&lt;&lt; activity-2 8 Knitting is so stressful 9 ===== 10 Knitting helps me relax 11 &gt;&gt;&gt;&gt; main 12</pre>					



## Prompt: GitHub branch and merge

# Fork

A screenshot of the GitHub 'Create a new fork' interface. The top bar shows the repository 'sandhikagalih / wpu-resolusi'. Below it, the navigation bar includes 'Code' (selected), 'Pull requests 155', 'Actions', 'Projects', 'Security', and 'Insights'. The main section is titled 'Create a new fork'. It explains that a fork is a copy of a repository and allows experimentation without affecting the original project. It shows the 'Owner' as 'fangelineap' and the 'Repository name' as 'wpu-resolusi', with a note that 'wpu-resolusi is available'. A checkbox for 'Copy the master branch only' is checked. A note at the bottom states: 'By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.' A 'Description (optional)' field contains the text 'Repository untuk menyimpan rencana WPU kedepannya'. A note at the bottom right says: 'You are creating a fork in your personal account.'

Digunakan untuk melakukan duplikasi repository milik orang lain lengkap dengan history commitnya dan meletakkannya pada repository kita dengan masih memiliki reference pada repository asli

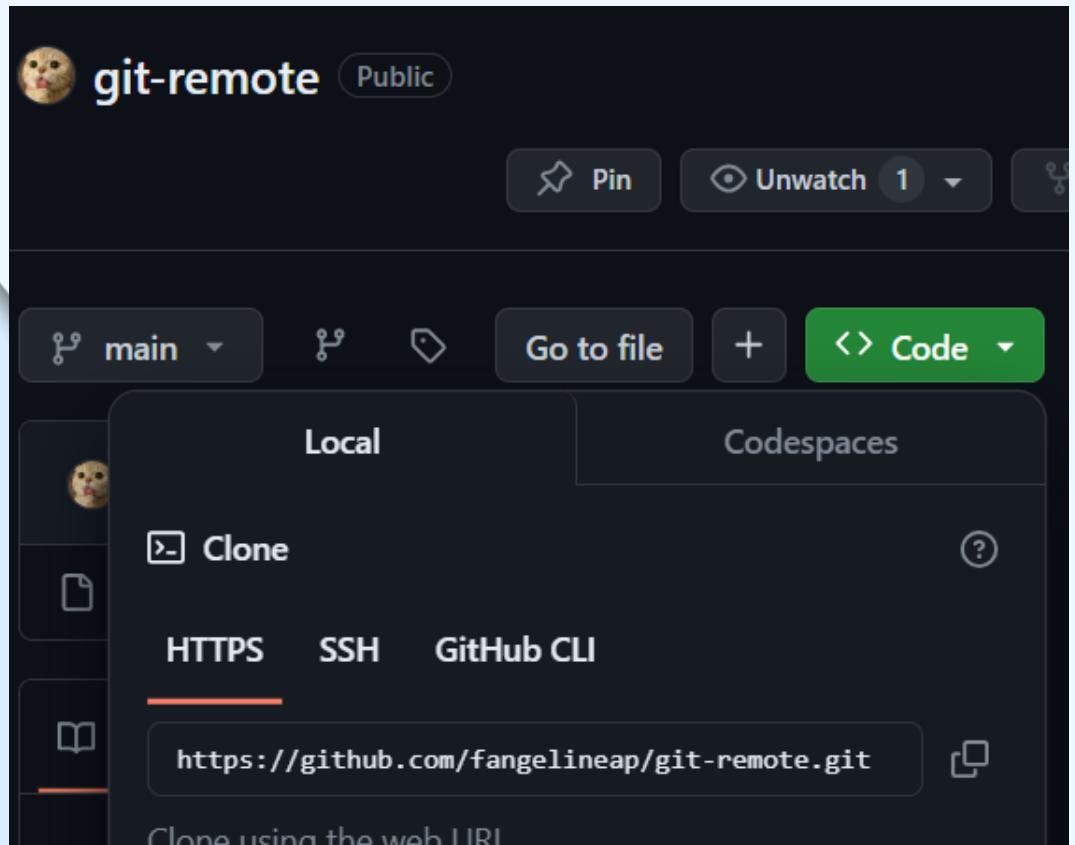
A screenshot of the GitHub 'Comparing changes' interface. It compares the 'base repository: sandhikagalih/wpu-resolusi' with the 'head repository: fangelineap/wpu-resolusi' and 'compare: master'. It shows that the branches are 'Able to merge'. The comparison summary indicates 2 commits, 2 files changed, and 1 contributor. A note says: 'Discuss and review the changes in this comparison with others. [Learn about pull requests](#)'. A green button says 'Create pull request'. Below, a list of commits shows two recent ones: 'Membuat file fei-resolusi.txt' by fangelineap committed 2 minutes ago, and 'Menambahkan node.js' by fangelineap committed now.

Prompt: Fork in GitHub

# Remote

Dilakukan dengan cloning repository dari github dan membuatnya menjadi remote repository sehingga kita dapat melakukan perubahan di lokal dan mengirimkannya pada remote tersebut (bisa disimpan di github, gitlab, dll)

Git clone (terhubung secara remote):



```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub
$ git clone https://github.com/fangelineap/git-remote.git
Cloning into 'git-remote'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub
$ cd git-remote

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote (main)
$ git remote
origin
```

Prompt: Remote repository

# Remote-2

Git push untuk melakukan commit pada remote repository.

Jika repository dari local:

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2
$ ls
index.php

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2
$ git init
Initialized empty Git repository in D:/Binus/Intern/Learning Materials/GitGithub/git-remote2/.git/
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git remote add origin https://github.com/fangelineap/git-remote2.git

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git remote
origin
```

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git add .

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git commit -m 'menambahkan file index.html'
[main 32f8a75] menambahkan file index.html
 1 file changed, 11 insertions(+)
   create mode 100644 index.html

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 20 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 451 bytes | 451.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/fangelineap/git-remote2.git
 d63f178..32f8a75 main -> main
```

# Remote Conflict

Terjadi ketika terdapat perubahan pada commit yang sudah terjadi di github dan local pada file yang sama

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git push
To https://github.com/fangelineap/git-remote2.git
 ! [rejected]      main -> main (fetch first)
error: failed to push some refs to 'https://github.com/fangelineap/git-remote2.git'
hint: Updates were rejected because the remote contains work that you do
hint: not have locally. This is usually caused by another repository pushing
hint: to the same ref. You may want to first integrate the remote changes
hint: (e.g., 'git pull ...') before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git fetch
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 994 bytes | 331.00 KiB/s, done.
From https://github.com/fangelineap/git-remote2
  caca69..3287bd5  main      -> origin/main
```

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ graph
* 369d4ce (HEAD -> main) menambahkan file style.css
| * 3287bd5 (origin/main) Create style.css
|/
* caca69 mengubah index.php

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/git-remote2 (main)
$ git pull origin main
From https://github.com/fangelineap/git-remote2
 * branch            main       -> FETCH_HEAD
Auto-merging style.css
CONFLICT (add/add): Merge conflict in style.css
Automatic merge failed; fix conflicts and then commit the result.
```

# Multiple Remote

Biasanya digunakan saat sebuah tim bekerja pada repository yang berbeda-beda (tidak disarankan) dan kita ingin melakukan fork lalu clone pada local.

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub  
$ cd simple-landing-page  
  
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/simple-landing-page (master)  
$ git remote  
origin  
  
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/simple-landing-page (master)  
$ git remote -v  
origin https://github.com/fangelineap/simple-landing-page.git (fetch)  
origin https://github.com/fangelineap/simple-landing-page.git (push)  
  
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/simple-landing-page (master)  
$ git remote add fei https://github.com/sandhikagalih/simple-landing-page.git  
  
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/simple-landing-page (master)  
$ git remote  
fei  
origin
```

Steps:

- Fork repository yang diinginkan
- Clone ke local repository
- Tambahkan remote baru agar selalu up to date dengan repo awal yang difork
- Gunakan git fetch pada remote sandhikagalih (repo asli) untuk melihat apakah terdapat commit baru di repo asli
- Jika iya, maka merge commit yang ada

```
fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/simple-landing-page (master)  
$ git remote -v  
origin https://github.com/fangelineap/simple-landing-page.git (fetch)  
origin https://github.com/fangelineap/simple-landing-page.git (push)  
sandhikagalih https://github.com/sandhikagalih/simple-landing-page.git (fetch)  
sandhikagalih https://github.com/sandhikagalih/simple-landing-page.git (push)
```

Prompt: Remote repository

# GitHub Pages

Hosting web secara gratis dengan menggunakan GitHub.



## Tips:

Pastikan terdapat file yang akan diakses pertama kali bernama index.html

The screenshot shows a GitHub repository page for `fangelineap.github.io`. The commit history is displayed, with the most recent commit being a rename of `index.html` 5 minutes ago. Other commits include creating `README.md` and initial commits for `header`, `sidebar`, `styles`, and `thumbnails`.

File	Commit Message	Time Ago
<code>index.html</code>	rename file	5 minutes ago
<code>README.md</code>	Create README.md	7 minutes ago
<code>thumbnails</code>	initial commit	15 minutes ago
<code>styles</code>	initial commit	15 minutes ago
<code>sidebar</code>	initial commit	15 minutes ago
<code>header</code>	initial commit	15 minutes ago

The screenshot shows a browser window displaying a cloned version of YouTube at `https://fangelineap.github.io`. The interface mimics the YouTube homepage, featuring a sidebar with links like Home, Explore, Subscription, Originals, YouTube Music, and Library. Below the sidebar, there are several video thumbnails with titles such as "Mini Birthday Cake Recipe | Baking With Gingersnap", "White Strawberry Cake Recipe", "Roll Cake Recipe", and "Carrot Cookies Recipe". Each video thumbnail includes a play button and a timestamp.

Prompt: GitHub Pages

# .gitignore

Digunakan untuk mengabaikan file tertentu agar tidak ikut ter-add pada saat melakukan commit.

Contoh: file system



## Tips

gitignore.io menyediakan list files yang harus di gitignore untuk setiap programming language

A screenshot of a file explorer window. At the top, there's a header with a magnifying glass icon and some text. Below it, a tree view shows a folder named 'GITIGNORE' containing several files and folders:

- A folder icon followed by 'data'
- A red arrow icon followed by '.gitignore'
- A blue document icon followed by 'config.txt'
- A blue square icon followed by 'index.html'
- A yellow square icon followed by 'script.js'
- A blue square icon followed by 'style.css'

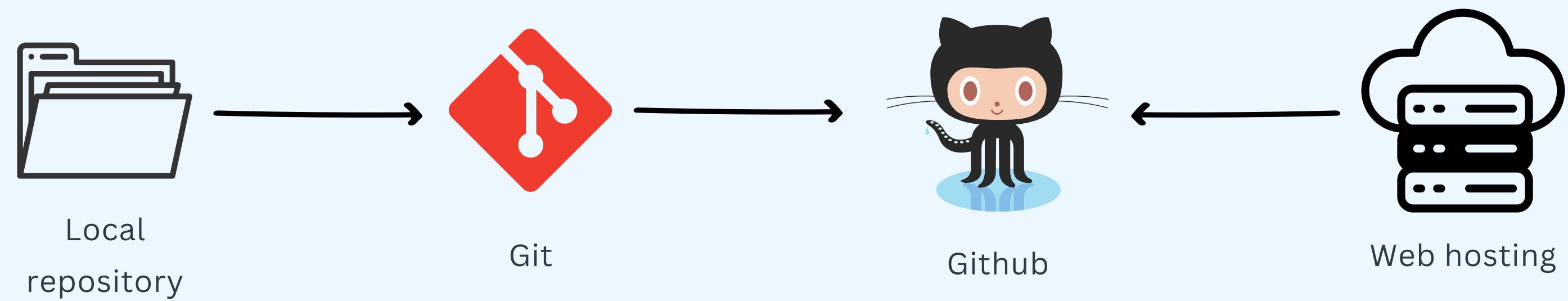
```
↳ .gitignore
1 # untuk mengabaikan file tertentu
2 config.txt
3
4 # untuk mengabaikan folder tertentu
5 data/
6
7 # untuk mengabaikan semua file yang memiliki extension .exe
8 *.exe

fange@LAPTOP-3AIV3NUB MINGW64 /d/Binus/Intern/Learning Materials/GitGithub/gitignore (main)
$ git status
On branch main
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   .gitignore
    new file:   script.js
```

Prompt: .gitignore file in Git

# Git & Web Hosting

Untuk melakukan web hosting kita dapat memanfaatkan git dengan menyambungkan local repository yang sudah dicommit dengan github dan web hosting akan pull update dari github berdasarkan commit terakhir yang sudah di approve di repository github.



Prompt: Git vs traditional way

# GitHub Repository

- ❖ Init, branch, dan merge: <https://github.com/fangelineap/yapping-time>
- ❖ Remote repository: <https://github.com/fangelineap/git-remote>
- ❖ Fork: <https://github.com/fangelineap/wpu-resolusi>
- ❖ GitHub Pages: <https://github.com/fangelineap/fangelineap.github.io>
- ❖ .gitignore: <https://github.com/fangelineap/coba-gitignore>



Thank  
You