

TUBES JARKOM

Nama: Muhammad Farhan Rosyadi

Kelas: D3SI 48-02

Nim: 607012400113

Nama Domain: github.io

Subdomain: muhammadfarhanrosyadi





Email yang digunakan: muhammadfarhanr09@gmail.com(gitbash)

Alamat url web: Link Web:

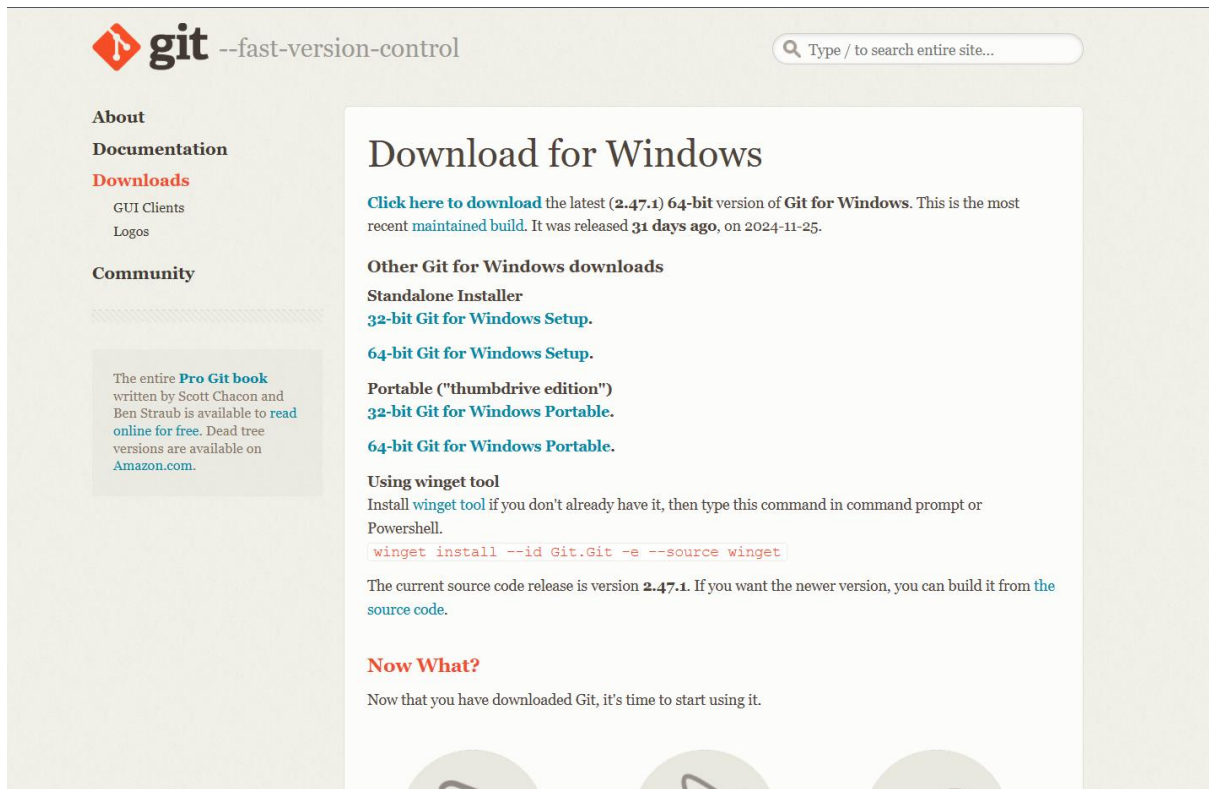
<https://muhammadfarhanrosyadi.github.io/tubes-jarkom-ros/>

Langkah Langkah Hosting Web

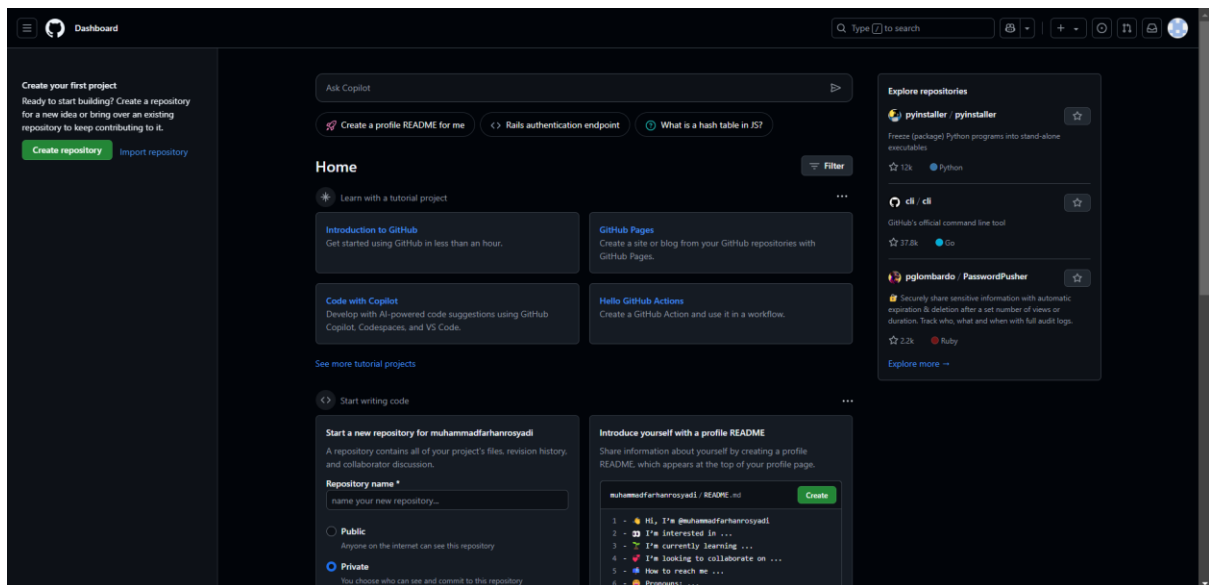
Langkah Pertama, Siapkan folder yang berisikan file untuk di hosting

 hosting	26/12/2024 17:53	Microsoft Edge HT...	3 KB
 IMG_20240814_170910	14/08/2024 17:09	JPG File	384 KB
 IMG20240508121339	08/05/2024 12:13	JPG File	3.092 KB
 style	26/12/2024 15:52	CSS Source File	1 KB

Langkah Kedua, Siapkan Aplikasi Gitbash untuk melakukan Setup hostingnya, Link Download: <https://git-scm.com/downloads>



Langkah Ketiga, buka github di web browser, lalu klik “Create Repository”



Langkah Keempat, Setelah mengklik “Create Repository” maka akan muncul tampilan seperti ini


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 *


Repository name *

 muhammadfarhanrosyadi ▾


 /

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

Description (optional)

☒  **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.](#)

 You are creating a public repository in your personal account.

Create repository


Langkah Kelima, Tambahkan nama pada kolom Repository Name, dan Click “Create 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 *

 muhammadfarhanrosyadi ▾

 /


Repository name *

tubes-jarkom-ros


✔ tubes-jarkom-ros is available.

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

Description (optional)

☒  **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

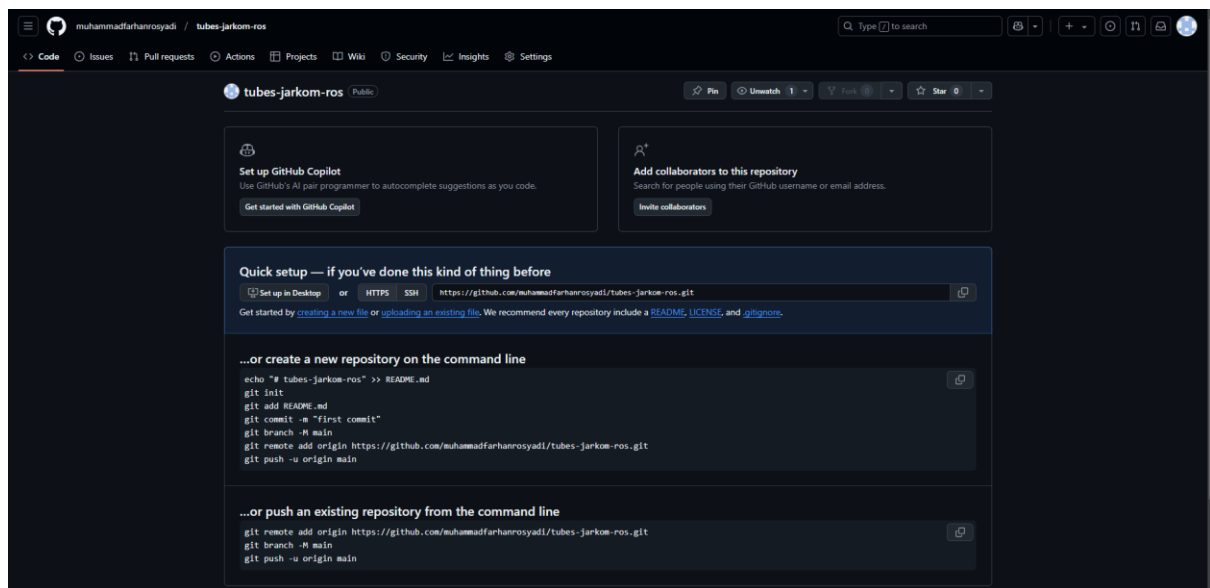
License: **None** ▾

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

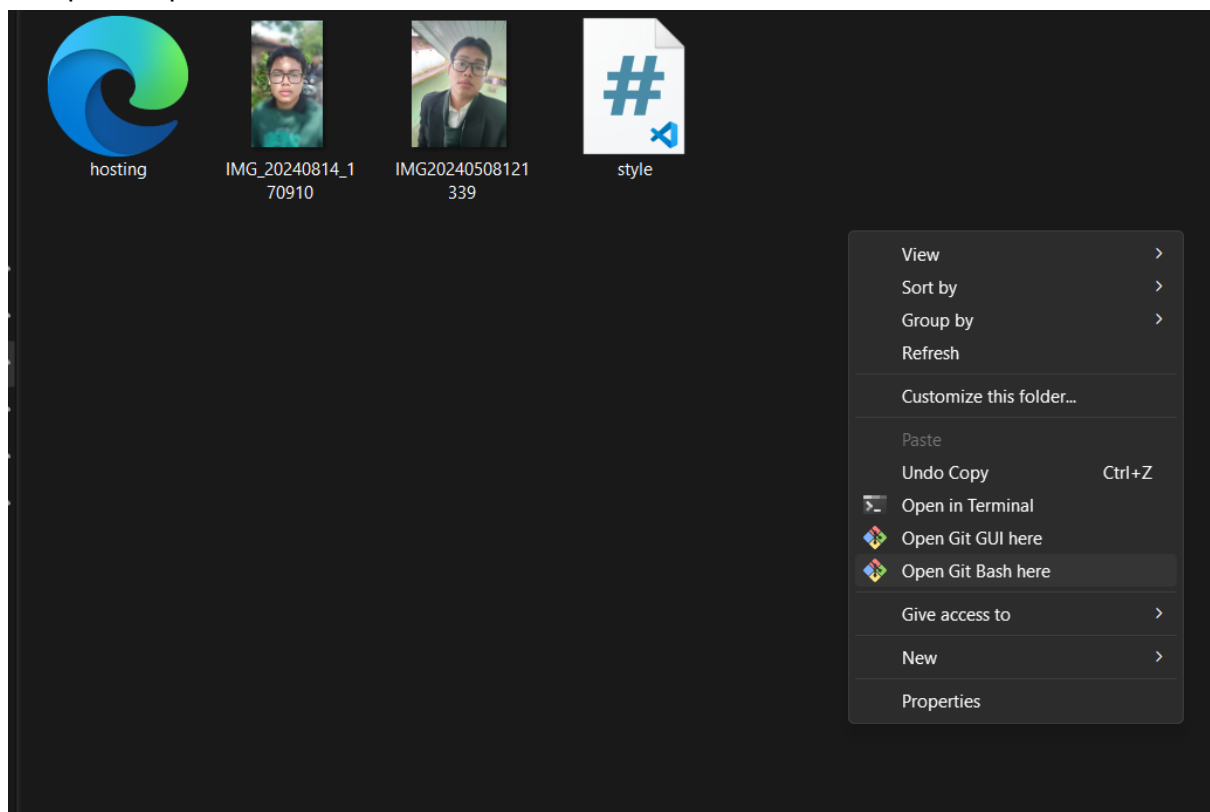
 You are creating a public repository in your personal account.

Create repository

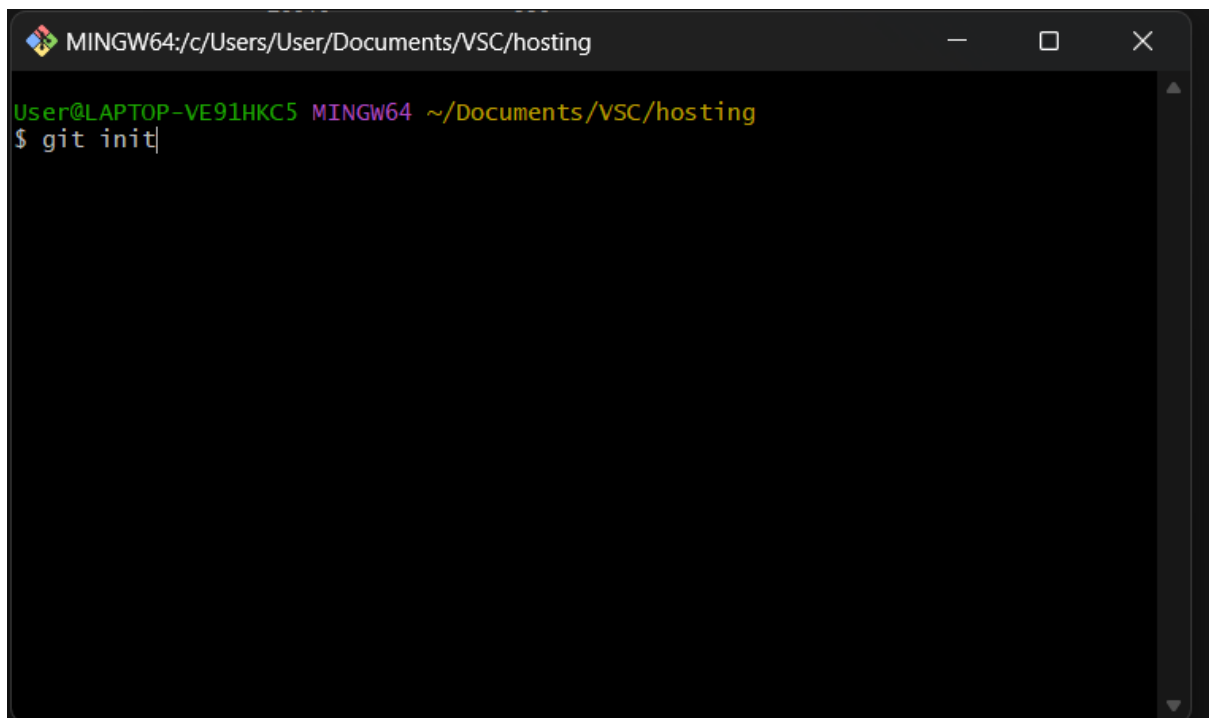
Langkah Keenam, Setelah Create Repository maka akan muncul tampilan seperti



Langkah Ketujuh, Pada folder yang ingin dihosting, click kanan lalu akan muncul pilihan, dan pilih “Open Git Bash Here”

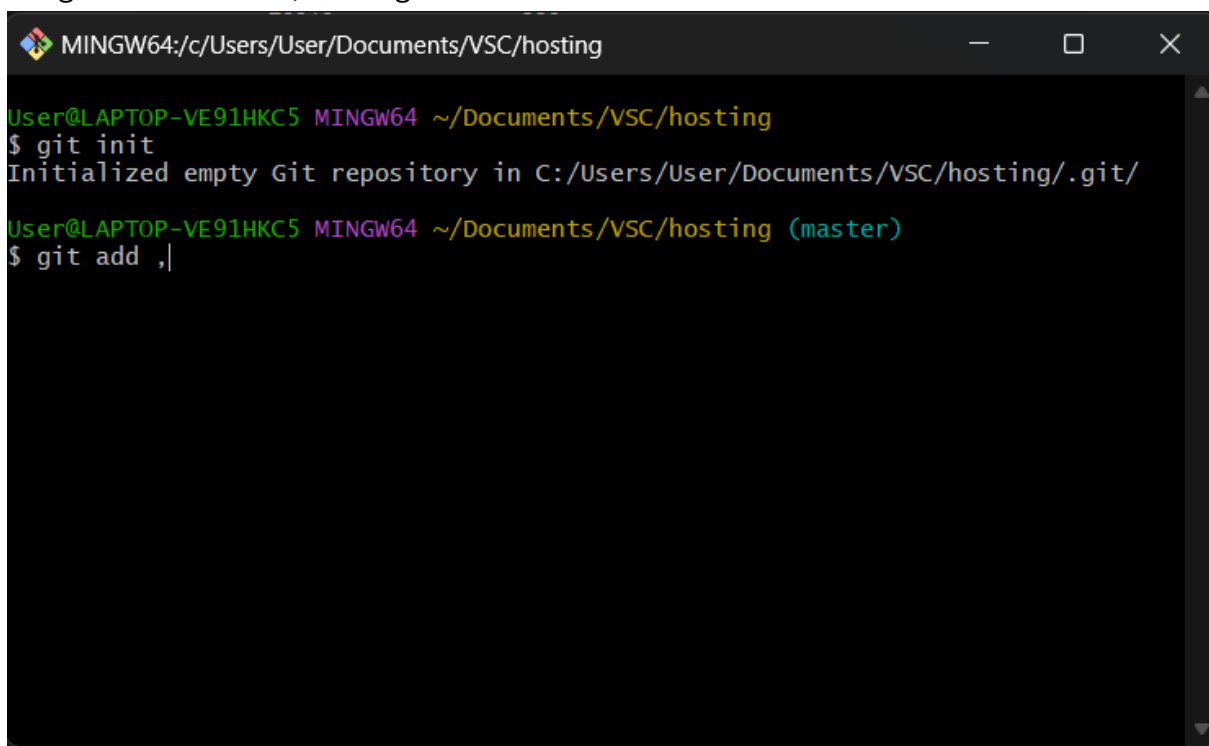


Langkah Kedelapan, Ketik “git init” kemudian enter

A terminal window titled 'MINGW64:/c/Users/User/Documents/VSC/hosting'. The prompt is 'User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/hosting'. The command '\$ git init' has been entered and is waiting for input.

```
MINGW64:/c/Users/User/Documents/VSC/hosting
User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/hosting
$ git init
```

Langkah Kesembilan, Ketik “git add .” kemudian enter

A terminal window titled 'MINGW64:/c/Users/User/Documents/VSC/hosting'. The prompt is 'User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/hosting'. The command '\$ git init' has been executed, resulting in the message 'Initialized empty Git repository in C:/Users/User/Documents/VSC/hosting/.git/'. The prompt now shows '(master)'. The command '\$ git add .' has been entered and is waiting for input.

```
MINGW64:/c/Users/User/Documents/VSC/hosting
User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/hosting
$ git init
Initialized empty Git repository in C:/Users/User/Documents/VSC/hosting/.git/
User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/hosting (master)
$ git add .|
```

Langkah Kesepuluh, ketik “git commit -m “example”” disini tanda kutip bisa di isi sesuai dengan apa yang kalian inginkan

```
MINGW64:/c/Users/User/Documents/VSC/tubesJarkom

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom
$ git init
Initialized empty Git repository in C:/Users/User/Documents/VSC/tubesJarkom/.git/

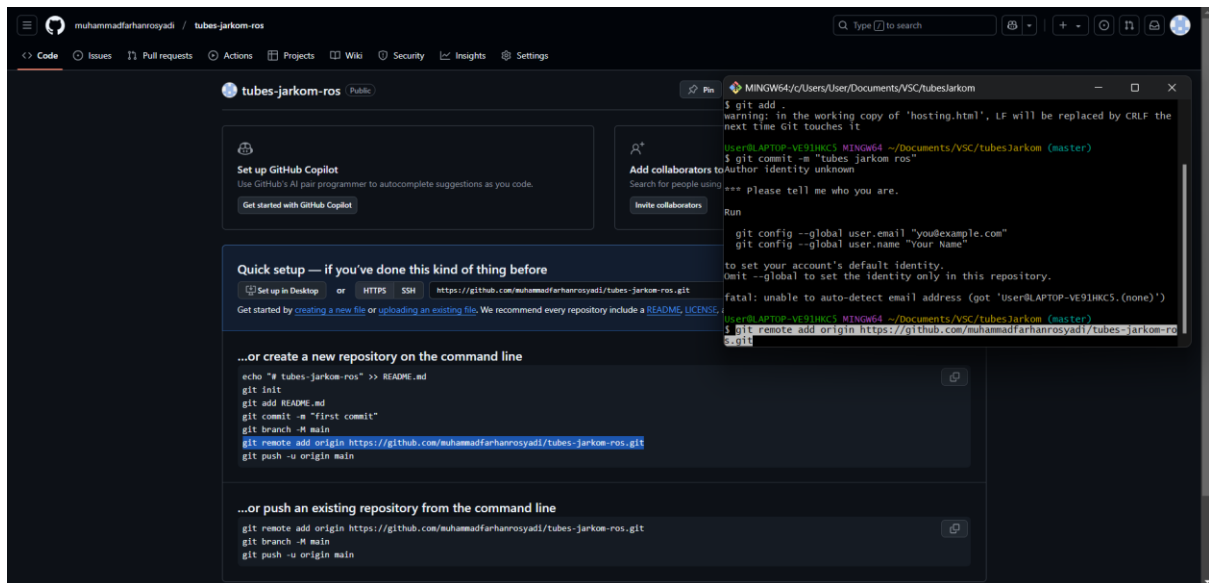
User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (master)
$ git add .
warning: in the working copy of 'hosting.html', LF will be replaced by CRLF the next time Git touches it

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (master)
$ git commit -m "tubes jarkom ros"
```

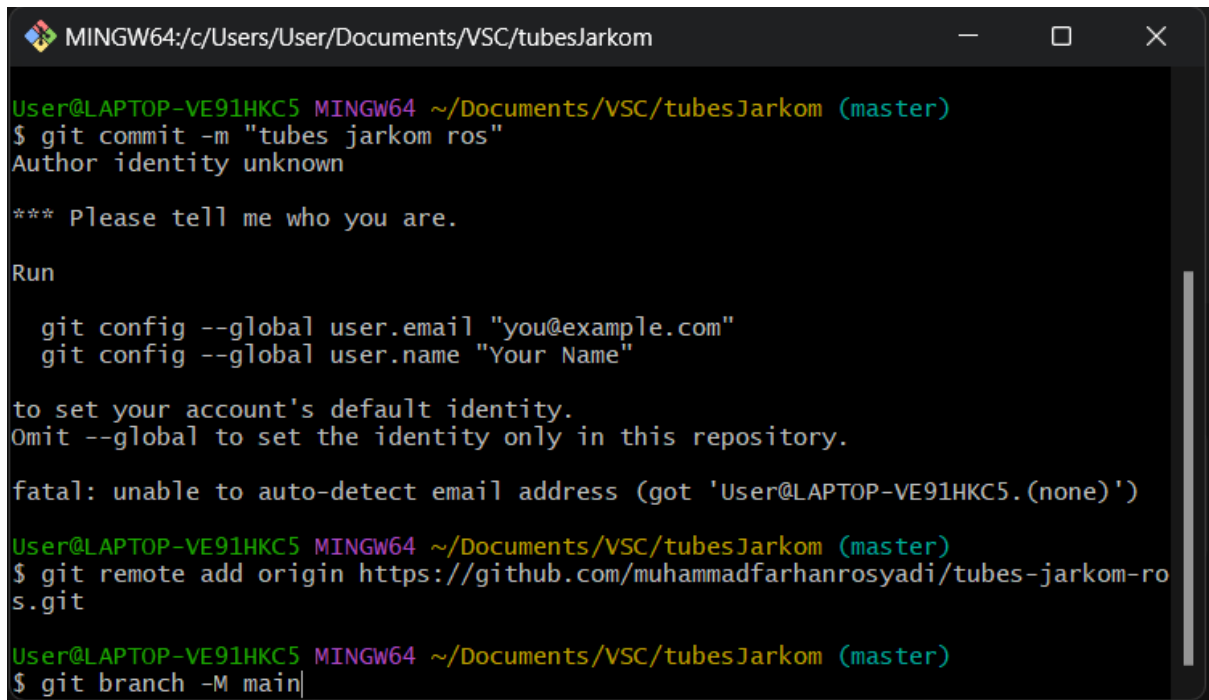
Langkah Kesebelas, pergi ke github, lalu salin “git remote origin (nama web kalian sesuai dengan yang sudah di auto generate github)”

The screenshot shows a GitHub repository page for 'tubes-jarkom-ros' by user 'muhammadfarhanrosyadi'. The page includes a 'Set up GitHub Copilot' section, a 'Quick setup' section with instructions for creating a new repository or pushing an existing one, and a 'Add collaborators' section. Overlaid on the right side of the page is a terminal window showing the execution of git commands: 'git add .', 'git commit -m "tubes jarkom ros"', and 'git remote add origin https://github.com/muhammadfarhanrosyadi/tubes-jarkom-ros.git'. The terminal also shows a warning about LF vs CRLF line endings and a fatal error about an undetectable email address.

Langkah Keduabelas, Tempel yang telah disalin pada Gitbash



Langkah Ketigabelas, ketik “git branch -M main” lalu enter



Langkah Keempatbelas ketik “git push -u” lalu enter

```
MINGW64:/c/Users/User/Documents/VSC/tubesJarkom
Author identity unknown
*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'User@LAPTOP-VE91HKC5.(none)')

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (master)
$ git remote add origin https://github.com/muhammadfarhanrosyadi/tubes-jarkom-ro
s.git

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (master)
$ git branch -M main

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (main)
$ git push -u|
```

Langkah Kelimabelas, Ketik “git push -u origin main” lalu ketik enter

```
MINGW64:/c/Users/User/Documents/VSC/tubesJarkom

fatal: unable to auto-detect email address (got 'User@LAPTOP-VE91HKC5.(none)')

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (master)
$ git remote add origin https://github.com/muhammadfarhanrosyadi/tubes-jarkom-ro
s.git

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (master)
$ git branch -M main

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (main)
$ git push -u
fatal: The current branch main has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin main

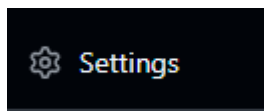
To have this happen automatically for branches without a tracking
upstream, see 'push.autoSetupRemote' in 'git help config'.

User@LAPTOP-VE91HKC5 MINGW64 ~/Documents/VSC/tubesJarkom (main)
$ git push -u origin main|
```

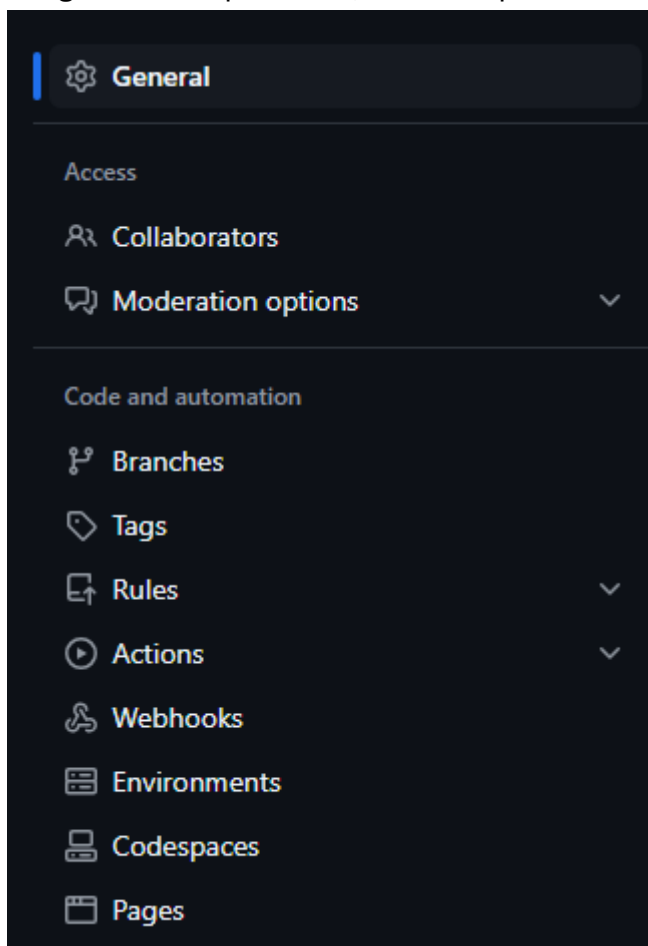
Langkah Keenambelas, Jika berhasil maka akan muncul tampilan seperti ini



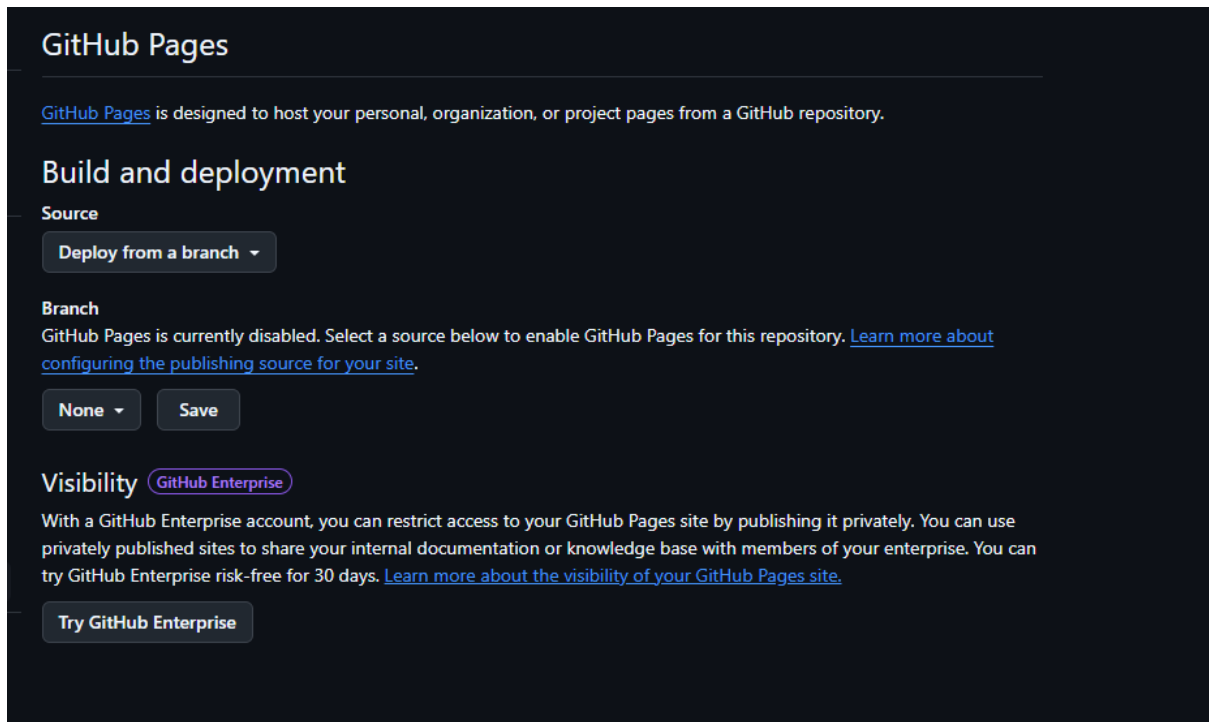
Langkah Ketujuhbelas, Click menu setting untuk mengatur



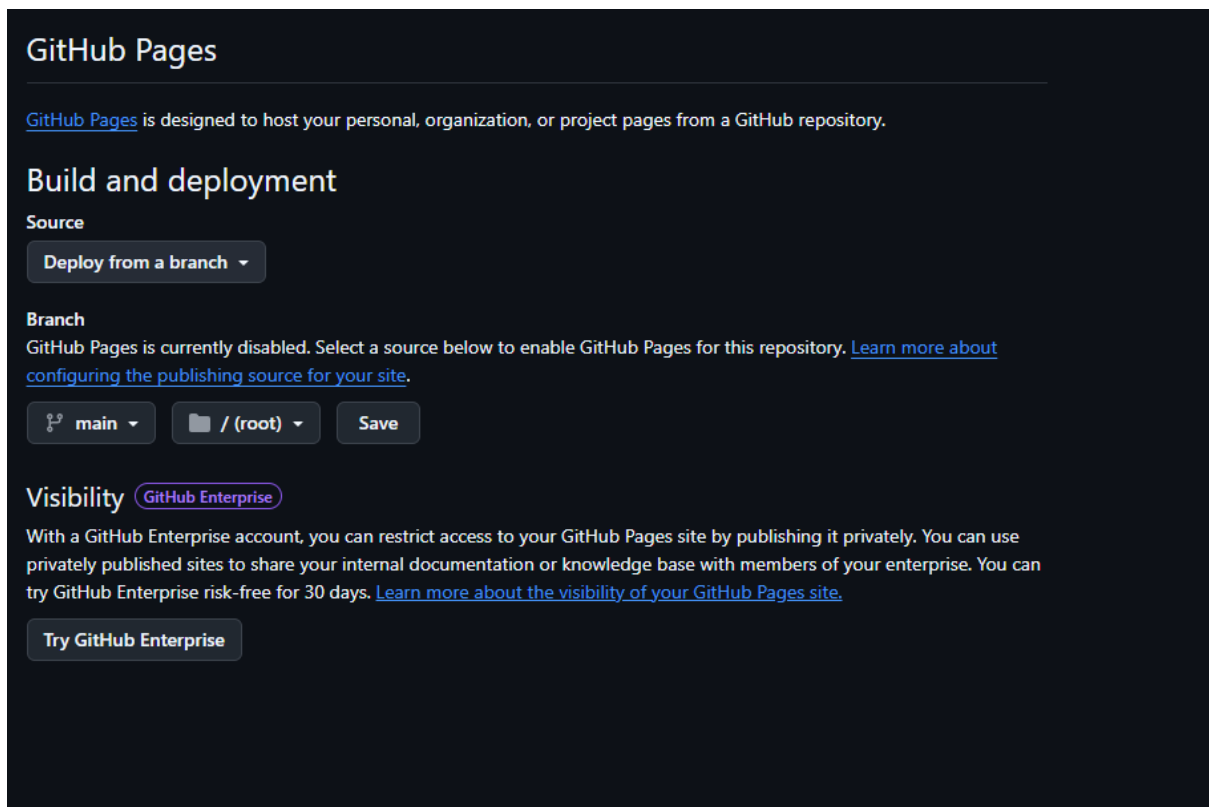
Langkah Kedelapanbelas, Pada tampilan setting yang seperti ini, pilih menu pages




Langkah Kesembilanbelas, pada pilihan Menu Branch click bagian “none”



Langkah Keduapuluh, ubah pilihan “None” menjadi “Main”



Lalu, click 

Langkah Keduapuluhsatu, Kita bisa mengatur Domain sesuai dengan apa yang kita inginkan, disini saya mengisi Domain dengan nama saya sendiri sesuai yang diminta pada tugas hosting ini, kemudian save

GitHub Pages

[GitHub Pages](#) is designed to host your personal, organization, or project pages from a GitHub repository.



Build and deployment

Source

Deploy from a branch ▾

Branch

Your GitHub Pages site is currently being built from the `main` branch. [Learn more about configuring the publishing source for your site.](#)

 main ▾  / (root) ▾ Save

Learn how to [add a Jekyll theme](#) to your site.

Custom domain

Custom domains allow you to serve your site from a domain other than `muhammadfarhanrosyadi.github.io`. [Learn more about configuring custom domains.](#)

Save Remove

☒ **Enforce HTTPS**

— Required for your site because you are using the default domain (`muhammadfarhanrosyadi.github.io`)

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site. When HTTPS is enforced, your site will only be served over HTTPS. [Learn more about securing your GitHub Pages site with HTTPS.](#)

Visibility GitHub Enterprise

With a GitHub Enterprise account, you can restrict access to your GitHub Pages site by publishing it privately. You can use privately published sites to share your internal documentation or knowledge base with members of your enterprise. You can try GitHub Enterprise risk-free for 30 days. [Learn more about the visibility of your GitHub Pages site.](#)

Try GitHub Enterprise

Jika sudah melakukan semua Langkah-langkah diatas maka web akan bisa di akses oleh semua orang melalui internet