



git




GitHub

O que é GIT?

- Git é um Sistema open-source de gerenciamento de versões, rápido e escalável.
- Foi desenvolvido pelo Linus Torvalds (criador do Linux) – Criado em 2005 pela necessidade de ter um software capaz de controlar a versão do Kernel Linux.
- Pode ser usado para controlar versões de qualquer formato (geralmente arquivos de texto)
 - Códigos fonte
 - Projetos
 - Documentação
 - Rastreamento de problemas
 - Histórico de alterações no código



Download

 **git** --distributed-even-if-your-workflow-isnt

Search entire site...

About
Documentation
Downloads
GUI Clients
Logos
Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

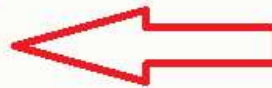
Download for Windows

[Click here to download](#) the latest (**2.39.1**) **64-bit** version of **Git for Windows**. This is the most recent [maintained build](#). It was released **9 days ago**, on 2023-01-17.

Other Git for Windows downloads

Standalone Installer
[32-bit Git for Windows Setup.](#)
[64-bit Git for Windows Setup.](#)
Portable ("thumbdrive edition")
[32-bit Git for Windows Portable.](#)
[64-bit Git for Windows Portable.](#)

Using winget tool
Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.



GIT Hub

- O GitHub é uma plataforma de hospedagem de código-fonte. Com ele, é possível desenvolver projetos de maneira colaborativa, num ambiente que muito lembra uma rede social.
- Nele, os desenvolvedores e programadores compartilham projetos para que o código seja desenvolvido por pessoas de toda a parte do mundo.
- A plataforma foi comprada pela Microsoft.
- É a rede social preferida dos desenvolvedores e programadores.



Principais termos

- **Repository:** O repositório é a pasta do projeto. Todo repositório tem uma pasta oculta `.git`. Isso é o que mostra para o git e para você que existe um repositório naquela pasta.
- **Clone:** clona-se esse repositório para o seu computador, trabalha-se nele, e então é pedida a permissão para atualizar as alterações online.
- **Commit:** Um commit é um grupo de alterações no código. Toda vez que você quiser "salvar" as alterações feitas por você no repositório, você commita essas mudanças.
- **Branch:** são separações de código. O branch padrão do projeto é o master. Branches normalmente são utilizados para separar alterações grandes ou novas funcionalidades do projeto



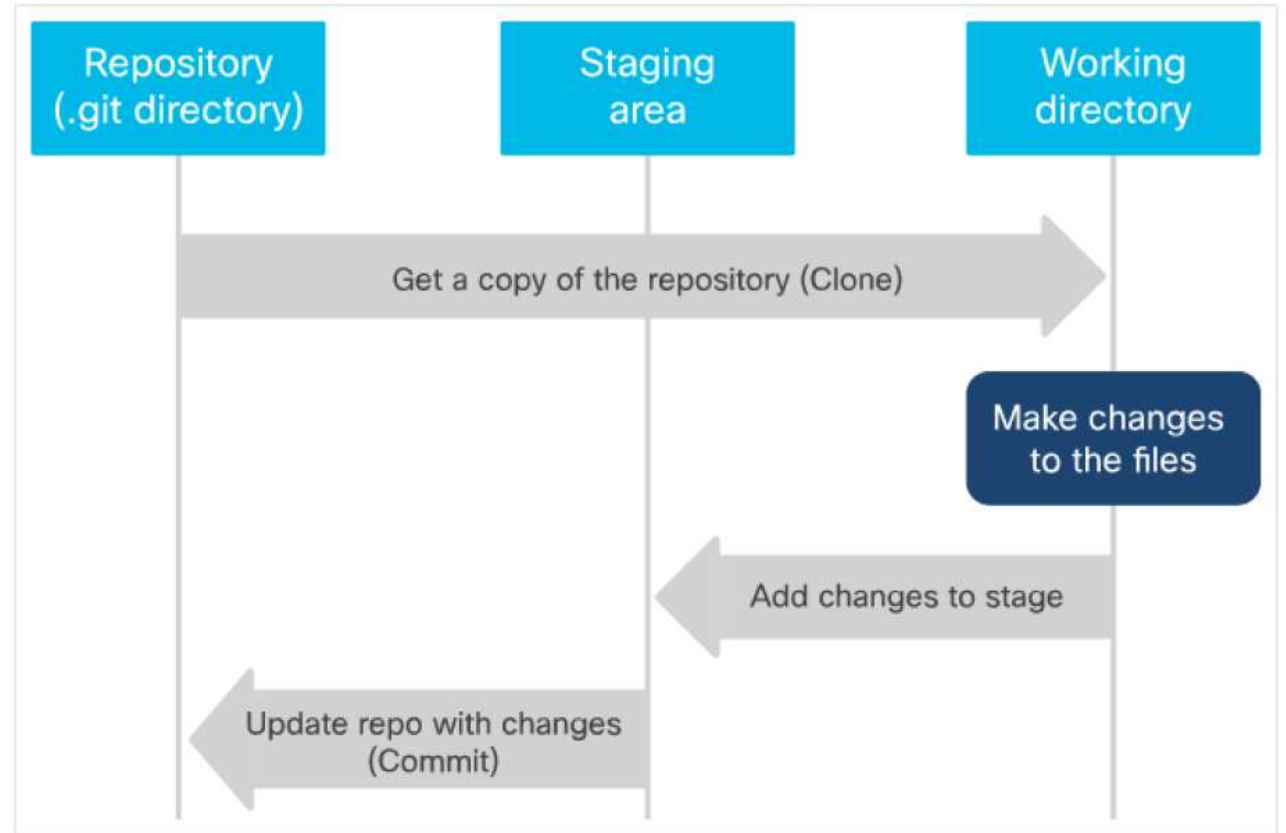
Principais termos

- **Merge:** Um merge é a união de duas branches, normalmente, merges são feitos na branch master.
- **Fork:** O fork é como um clone, porém dentro do github. Isso quer dizer que o repositório não vai ser baixado para seu computador, mas será criado um igual na sua conta.
- **Pull:** O comando pull é usado para 'puxarmos' as atualizações, ou seja, para capturarmos as atualizações feitas no repositório remoto de origem e trazermos para nosso ambiente local.
- **Push:** O comando push é usado para enviar as atualizações para o servidor remoto
- **Stage:** o stage então agrupa todas as atualizações que serão enviadas no próximo commit

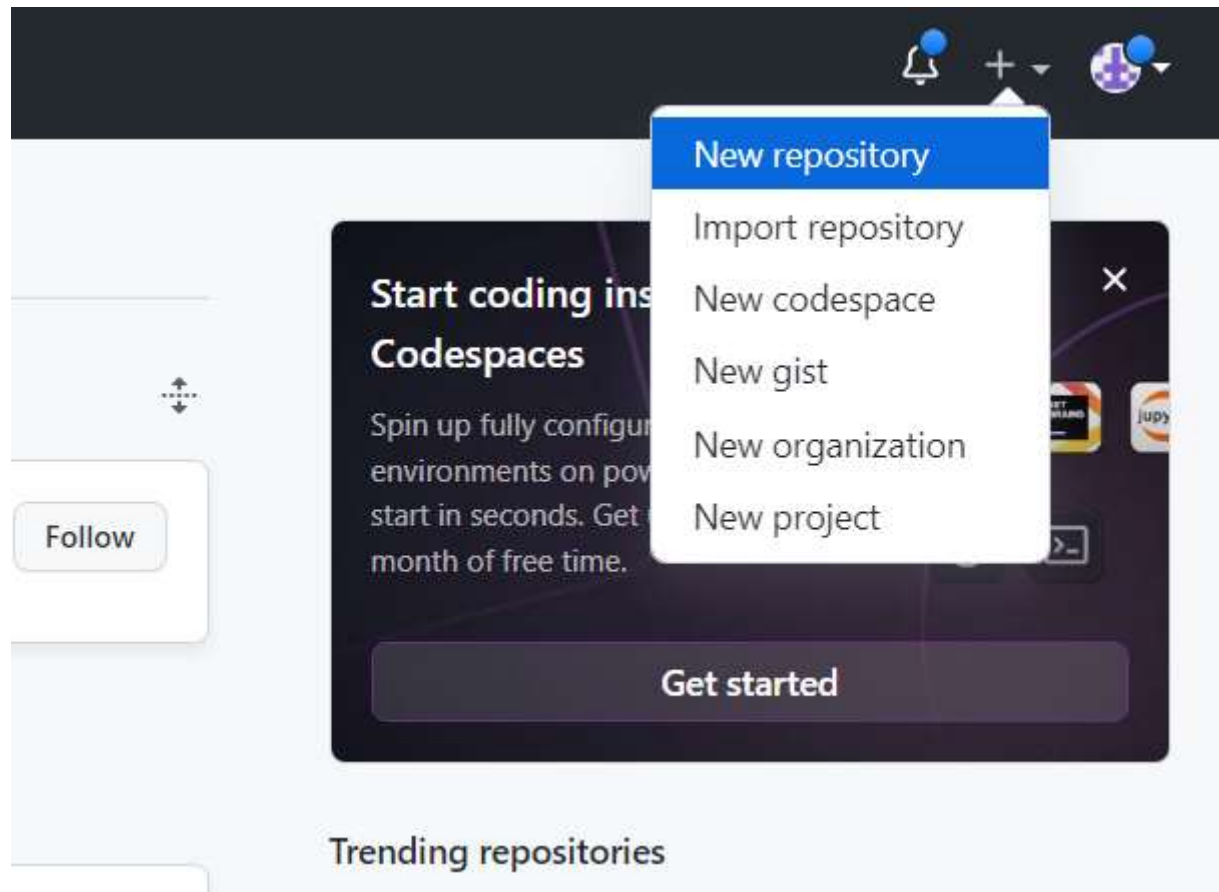


Git three stages

- The three stages are:
 - Repository (the .git directory)
 - Working directory
 - Staging area
- The three states are:
 - Committed
 - Modified
 - Staged



Create repository



Create a new repository on the command line



```
MINGW64:/c/Users/horlab302/pastaGIT
horlab302@HORLAB302-17776 MINGW64 ~
$ mkdir pastaGIT
horlab302@HORLAB302-17776 MINGW64 ~
$ cd pastaGIT/
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT
$ |
```

Execute the command `git init`

A screenshot of a Windows terminal window with a black background and white text. The window title bar at the top shows the icon for MINGW64, followed by the path 'MINGW64:/c/Users/horlab302/pastaGIT'. On the right side of the title bar are standard window controls: a minus sign, a square icon, and an 'X' icon. The terminal content shows a series of commands and their outputs. The first command is 'mkdir pastaGIT', which is executed successfully. The second command is 'cd pastaGIT/', also executed successfully. The third command is 'git init', which results in the output 'Initialized empty Git repository in C:/Users/horlab302/pastaGIT/.git/'. The final line shows the prompt '\$ |' with a vertical cursor, indicating the terminal is ready for the next command. The prompt text is color-coded: 'horlab302@HORLAB302-17776' is green, 'MINGW64' is purple, '~' is yellow, and '(master)' is cyan.

```
horlab302@HORLAB302-17776 MINGW64 ~  
$ mkdir pastaGIT  
  
horlab302@HORLAB302-17776 MINGW64 ~  
$ cd pastaGIT/  
  
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT  
$ git init  
Initialized empty Git repository in C:/Users/horlab302/pastaGIT/.git/  
  
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)  
$ |
```

Config the user

- git config user.name "Thiago Almeida"
- git config user.email "thiagoalmeida@live.com"

A screenshot of a Windows terminal window with a black background and white text. The title bar at the top shows the MINGW64 logo and the path 'MINGW64:/c/Users/horlab302/pastaGIT'. The terminal content shows three lines of commands and their prompts. The first line shows the prompt 'horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)' followed by the command '\$ git config user.name "Nome do Usuario"'. The second line shows the same prompt followed by '\$ git config user.email "usuario@gmail.com"'. The third line shows the same prompt followed by a single '\$' character, indicating the prompt is ready for input.

```
MINGW64:/c/Users/horlab302/pastaGIT

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git config user.name "Nome do Usuario"

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git config user.email "usuario@gmail.com"

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$
```

Create the SSH-Keygen

- Execute o comando ssh-keygen

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/horlab302/.ssh/id_rsa):
Created directory '/c/Users/horlab302/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/horlab302/.ssh/id_rsa
Your public key has been saved in /c/Users/horlab302/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:yj7msPrWOP736TIT49bCuLmn/5S9gmh2G3c9iRIXEm0 horlab302@HORLAB302-17776
The key's randomart image is:
+---[RSA 3072]-----+
|
| ..
| .E
| ...
| . .
| S . .
| . .0 = 0 .
| .00=.== + +
| ++0.&+0+ . .
| .+=B0@oX=. .
+---[SHA256]-----+
```

Copy the SSH-Keygen

```
MINGW64:/c/Users/horlab302/.ssh
horlab302@HORLAB302-17776 MINGW64 ~
$ cd ..

horlab302@HORLAB302-17776 MINGW64 /c/Users
$ cd horlab302/

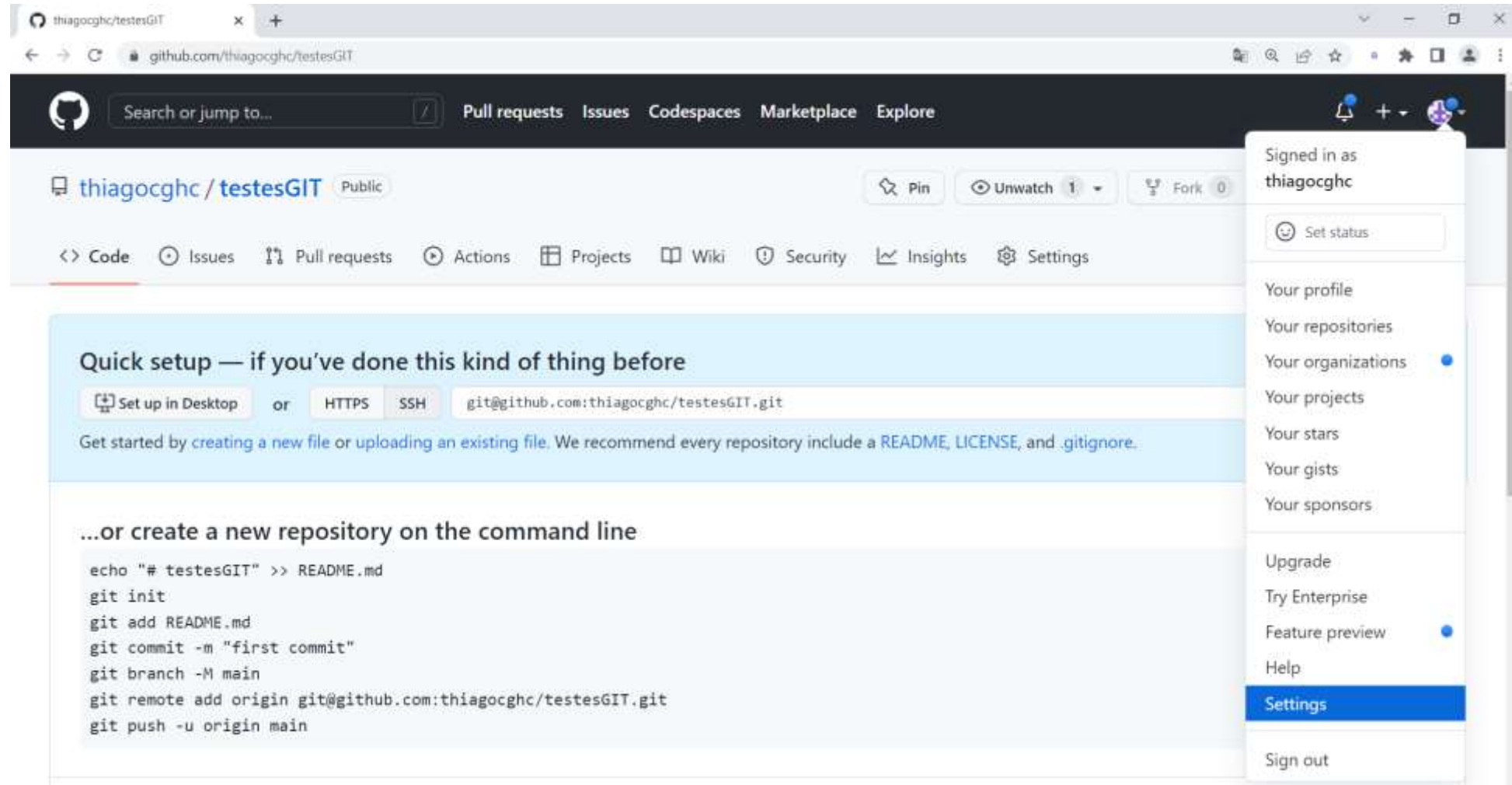
horlab302@HORLAB302-17776 MINGW64 ~
$ cd .ssh/

horlab302@HORLAB302-17776 MINGW64 ~/.ssh
$ ls
id_rsa  id_rsa.pub

horlab302@HORLAB302-17776 MINGW64 ~/.ssh
$ cat id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQgQDHVE1ZY33BJzwffutzJRCyiAj2K8ZcwmiD+kTtNiVL
XmQ1kC6WhlJ0HSm565wyboPENJYFzEu/x0jTc/Gjyv12FjA3C7Uv907boQCV0eb0TCekf79kErFyzRlp
/+jMc/HF8Zs0weDcUCJDSDIb/feHeUmxIk0YqFewGJ1q/fM+yEt1Q8vCTcyLlAA9P6TKoGBWuz1zVcrP
fZ1FwMamlaTB7IIdPZaAZjfmkL6wvYbbMBZBLNZiaROIHQRFD2qevoBZb37Zbxoup+XQnGcAzhX470ye
V6S10tSWS51OuEC5DB2KDc4CPURZtdrrVUDLM/cSwDpboux/JVEhLumqwqlTNhzN9194GXietY6gNCyC
d8gzZAWlB3jkx190ygvdnIgbQhGqB9CDSybQk0GRP9cg0miYW9BV9Y3lJx09FEpRvSP49xDg7inZQ5Ml
bcTaaZnQ2NrjjMOn9V5lgsVlsapsB5YO1tOuMvBW/HM1Ja5sCtKd/dgsz7L79f0aw1B4Lgk= horlab3
02@HORLAB302-17776

horlab302@HORLAB302-17776 MINGW64 ~/.ssh
$ |
```

Configure SSH and GPG keys



The screenshot shows the GitHub web interface for the repository `thiagocghc/testesGIT`. The user `thiagocghc` is signed in, and the user menu is open, showing options like `Set status`, `Your profile`, `Your repositories`, `Your organizations`, `Your projects`, `Your stars`, `Your gists`, `Your sponsors`, `Upgrade`, `Try Enterprise`, `Feature preview`, `Help`, `Settings` (highlighted), and `Sign out`.

Quick setup — if you've done this kind of thing before

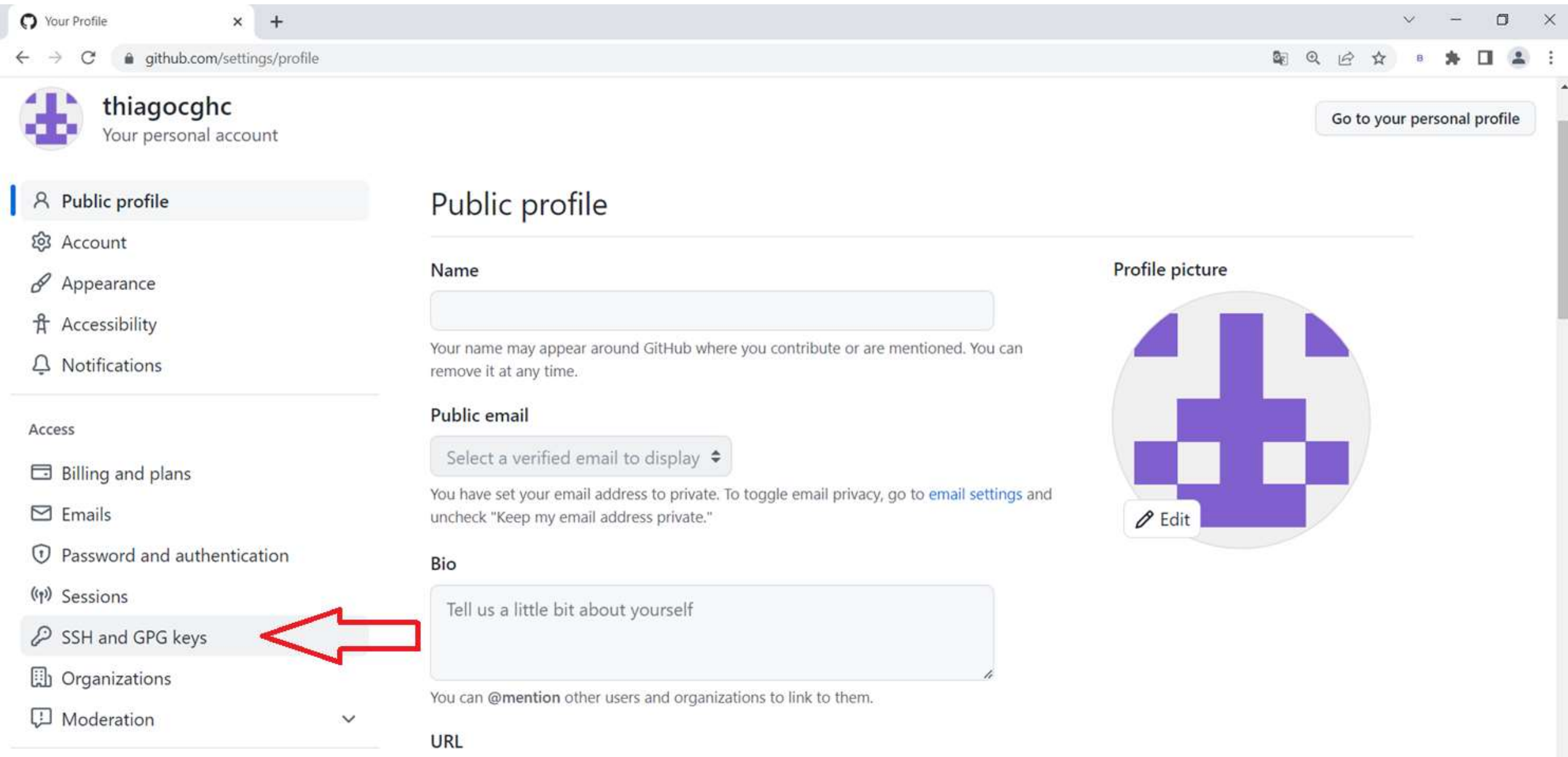
[Set up in Desktop](#) or [HTTPS](#) [SSH](#) `git@github.com:thiagocghc/testesGIT.git`

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line


```
echo "# testesGIT" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin git@github.com:thiagocghc/testesGIT.git
git push -u origin main
```

Configure SSH and GPG keys



Your Profile x +

github.com/settings/profile

 **thiagocghc**
Your personal account

[Go to your personal profile](#)

- Public profile
- Account
- Appearance
- Accessibility
- Notifications

Access

- Billing and plans
- Emails
- Password and authentication
- Sessions
- SSH and GPG keys**
- Organizations
- Moderation

Public profile

Name

Your name may appear around GitHub where you contribute or are mentioned. You can remove it at any time.

Public email

Select a verified email to display

You have set your email address to private. To toggle email privacy, go to [email settings](#) and uncheck "Keep my email address private."


Bio

Tell us a little bit about yourself

You can @mention other users and organizations to link to them.

URL

Profile picture



[Edit](#)

New SSH Key

The screenshot shows the GitHub web interface for a user named thiagocghc. The page is titled "SSH and GPG keys" and is located at `github.com/settings/keys`. The left sidebar contains navigation links for the user's profile, account settings, appearance, accessibility, notifications, and various access settings. The main content area is titled "SSH keys" and includes a "New SSH key" button. Below this, a list of existing SSH keys is shown, each with a key icon, a label, the SHA256 fingerprint, the date added, the last used date, and a "Delete" button.

SSH and GPG keys

thiagocghc
Your personal account

Go to your personal profile

Public profile
Account
Appearance
Accessibility
Notifications

Access




Billing and plans
Emails
Password and authentication
Sessions
SSH and GPG keys
Organizations

SSH keys

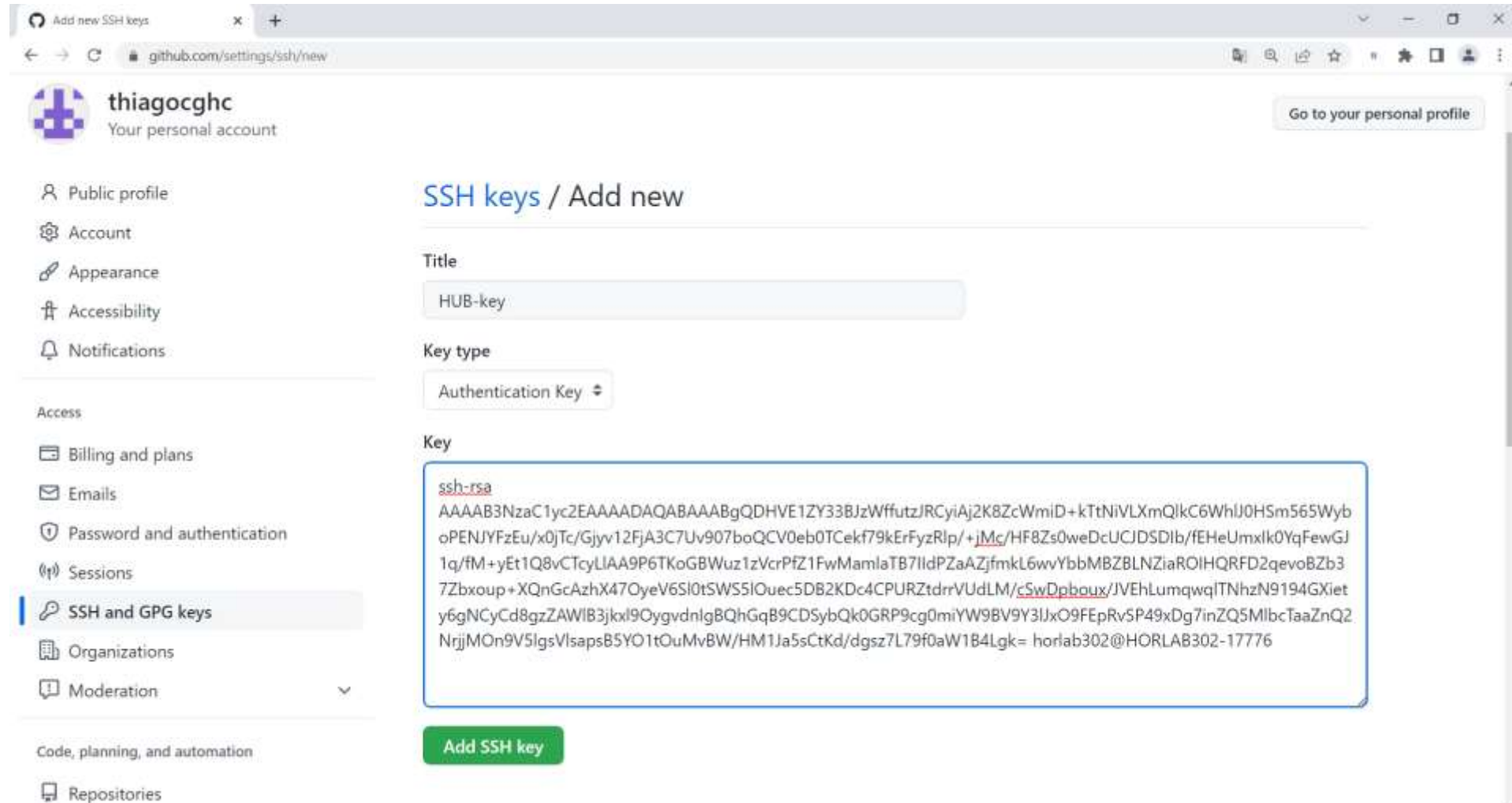
New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication Keys

	LinuxVM SHA256:zc3oi600wGcJxoYB+EsTdIC8DLygxxih1P8PiF4zqMI Added on Sep 8, 2022 Last used within the last 5 months — Read/write	Delete
	devnet-abredes SHA256:2NdpbJJ28LShqtSFD4bj6ca1ZTgF35FoI7ukjibui0 Added on Sep 10, 2022 Last used within the last 5 months — Read/write	Delete
	Windows SHA256:vq5m1nqTJwTS1EIHjSY2kmEP+wLCDK10nTr2UjXp1C0	Delete

Paste the SSH Key



thiagocghc
Your personal account

Public profile
Account
Appearance
Accessibility
Notifications

Access

Billing and plans
Emails
Password and authentication
Sessions
SSH and GPG keys
Organizations
Moderation

Code, planning, and automation

Repositories

SSH keys / Add new

Title

HUB-key

Key type

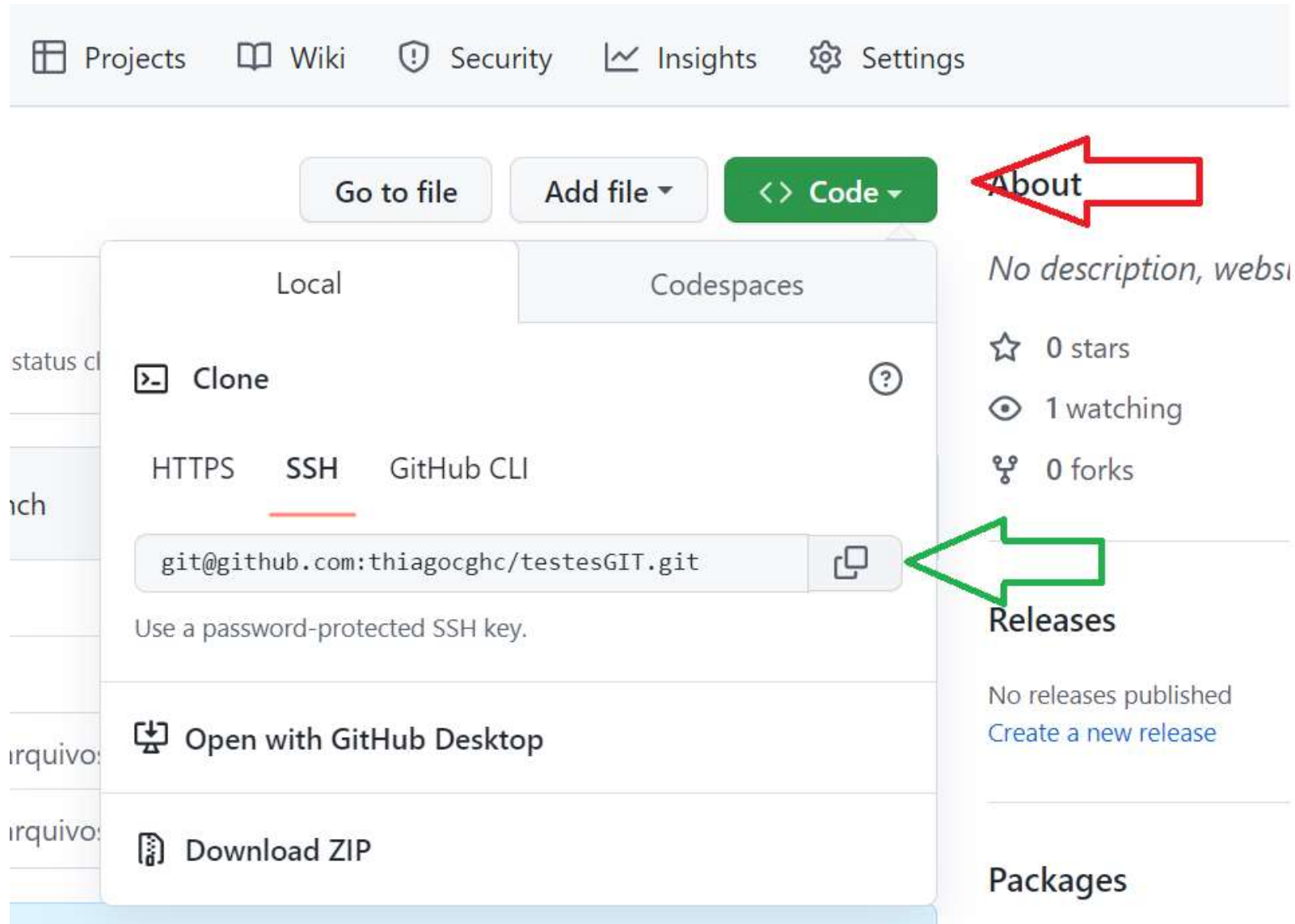
Authentication Key

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQGDHVE1ZY33BJzWffutzJRCyiAj2K8ZcWmiD+kTtNiVLXmQlkC6WhIJ0HSm565Wyb
oPENJYFzEu/x0jTc/Gjyv12FjA3C7Uv907boQCV0eb0TCekf79kErFyzRlp/+jMc/HF8Zs0weDcUCJDSdIb/fEHeUmxlk0YqFewGJ
1q/fM+yEt1Q8vCTcyLIAA9P6TKoGBWuz1zVcrPfZ1FwMamlaTB7IldPZaAZjfmkL6wvYbbMBZBLNZiaROIHQRFD2qevoBZb3
7Zbxoup+XQnGcAzhX47OyeV6SI0tSWSSlOuec5DB2KDc4CPURZtdrrVUdLM/cSwDpboux/JVEhLumqwqITNhzn9194GXiet
y6gNCyCd8gzZAWIB3jkl9OygvdnlgBQhGqB9CDsybQk0GRP9cg0miYW9BV9Y3IJxO9FEpRvSP49xDg7inZQ5MlbcTaaZnQ2
NrjjMOn9V5lgsVlsapsB5YO1tOuMvBW/HM1Ja5sCtKd/dgsz7L79f0aW1B4Lgk= horlab302@HORLAB302-17776
```

Add SSH key

git remote add



git remote add

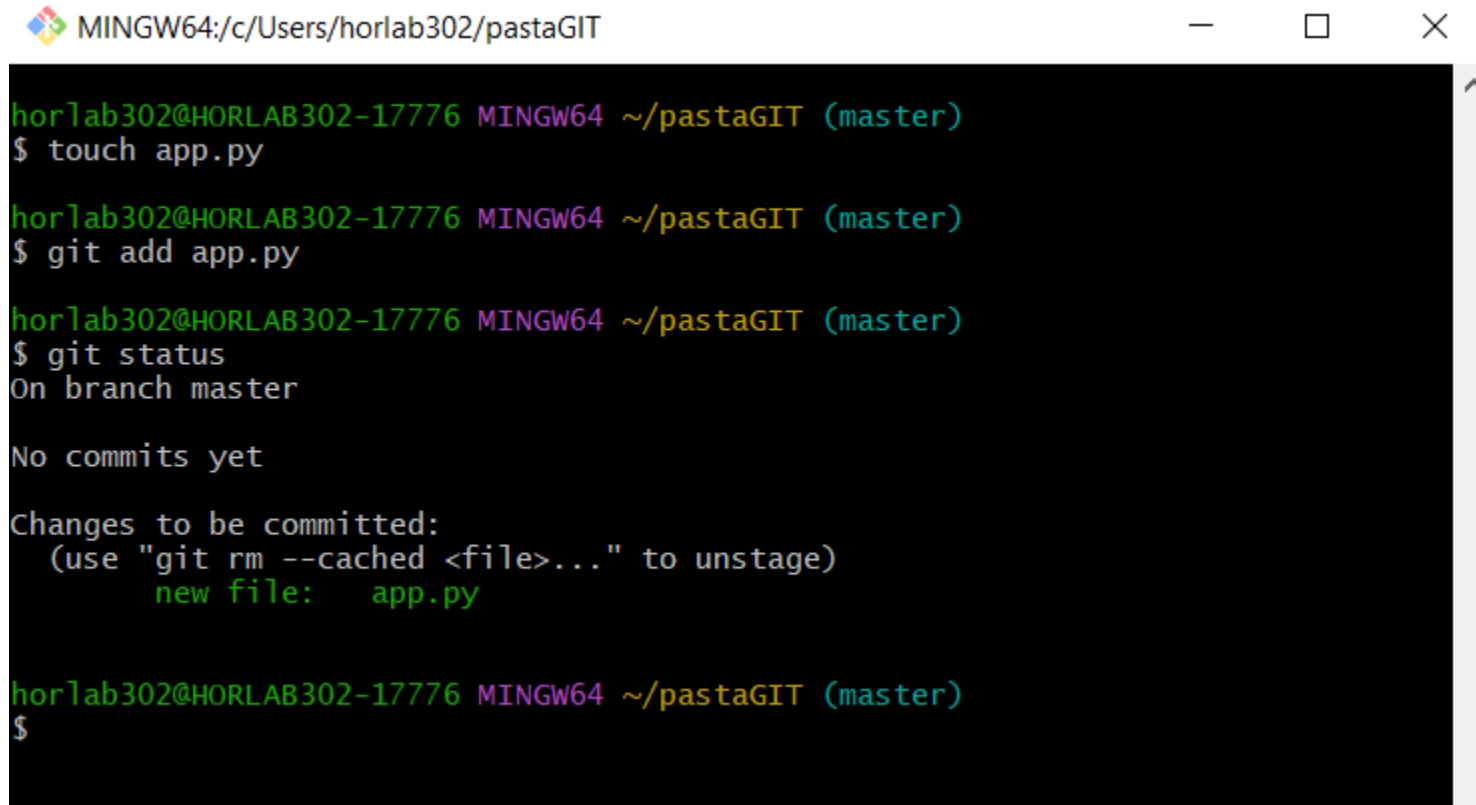
- Git remote add, sincroniza a pasta do projeto com o repositório remoto do github

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git status
On branch master
nothing to commit, working tree clean

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git remote add origin git@github.com:thiagocghc/testesGIT.git

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ |
```

New file



```
MINGW64:/c/Users/horlab302/pastaGIT

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ touch app.py

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git add app.py

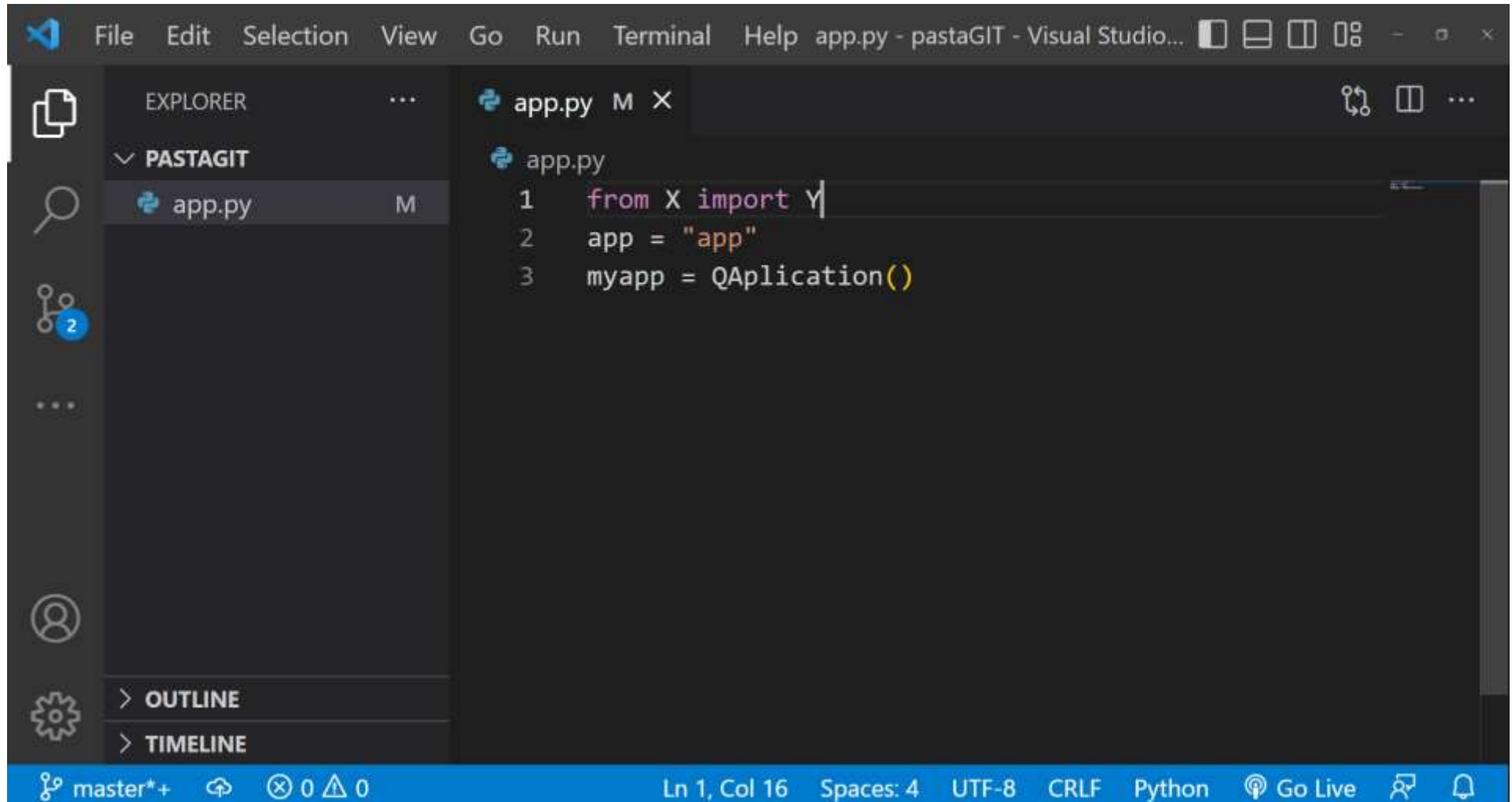
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git status
On branch master

No commits yet

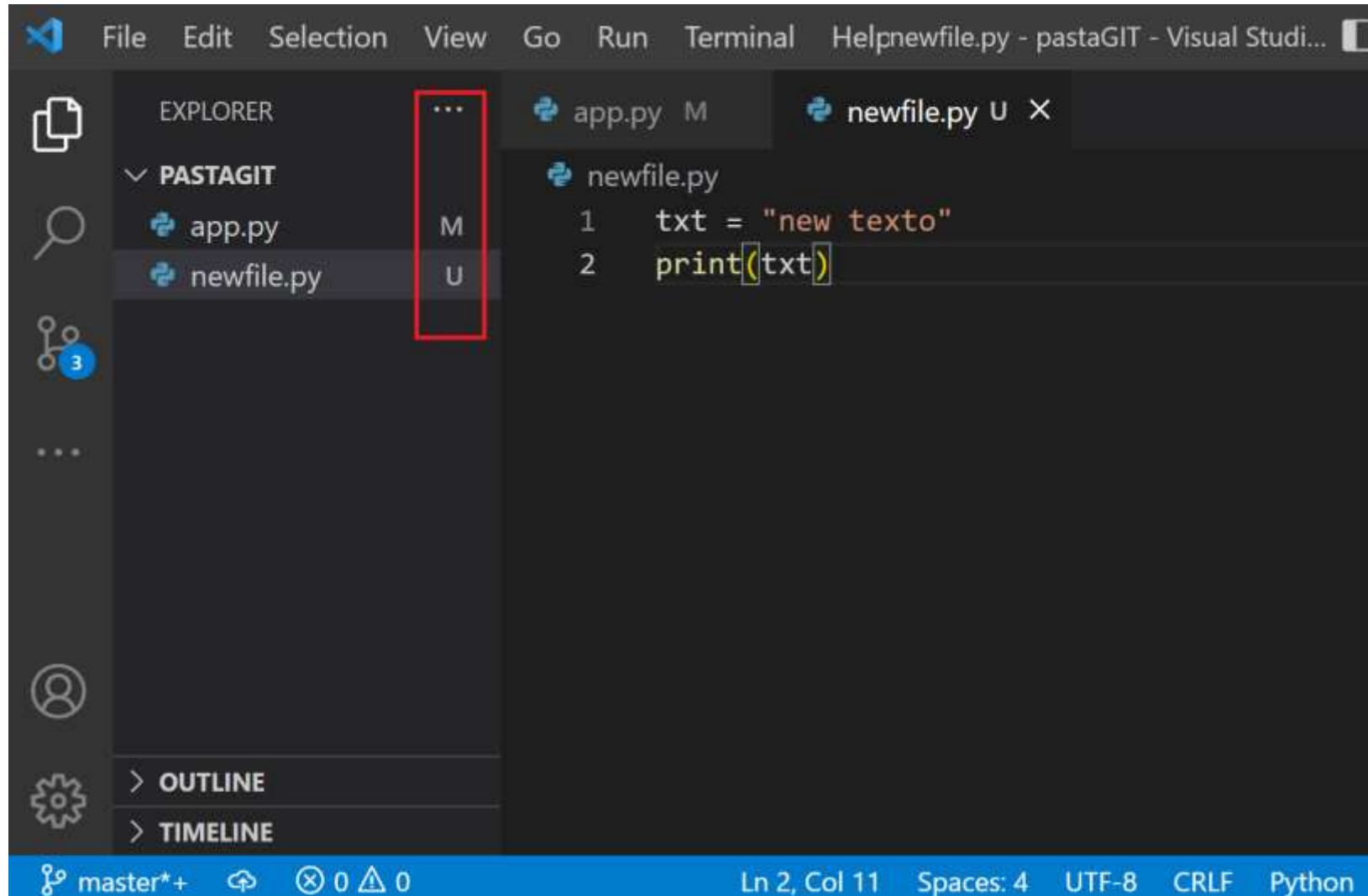
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   app.py

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$
```

New file



New file



git status

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git status
On branch master

No commits yet

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

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   app.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    newfile.py
```

git add

- Git add: para adicionar os arquivos na stage area

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git add app.py newfile.py

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git commit -a -m "My first commit"
```


git commit

- Git commit: para para salvar as informações
 - - a todos os arquivos
 - -m “mensagem de envio”

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git add app.py newfile.py

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git commit -a -m "My first commit"
```

git commit

- Git commit: para para salvar as informações
 - - a todos os arquivos
 - -m “mensagem de envio”

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git add app.py newfile.py

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git commit -a -m "My first commit"
```

git commit

- Git commit: para para salvar as informações
 - - a todos os arquivos
 - -m “mensagem de envio”

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git commit -a -m "My first commit"
[master (root-commit) 0a83266] My first commit
 2 files changed, 5 insertions(+)
 create mode 100644 app.py
 create mode 100644 newfile.py

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git status
On branch master
nothing to commit, working tree clean

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ |
```

git push

```
MINGW64:/c/Users/horlab302/pastaGIT
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git push origin master
The authenticity of host 'github.com (20.201.28.151)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvV6TuJJhbpZisF/zLDA0zPMSvHdk
r4UvCOqU.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])?
yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 327 bytes | 163.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:thiagocghc/testesGIT.git
 * [new branch]      master -> master
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ |
```

git push

- O git push origin main/master envia os arquivos para o repositório remoto do GIT

The screenshot shows the GitHub interface for a repository named 'testesGIT'. At the top, there's a header with the repository name and a 'Public' badge. Below this is a navigation bar with links for Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main content area shows the 'master' branch selected, with 1 branch and 0 tags. There are buttons for 'Go to file', 'Add file', and 'Code'. Below this, a commit history table is displayed, showing a single commit by 'miNombreDeUsuario' with the message 'My first commit'. The commit details show the commit hash '0a83266', the time '6 minutes ago', and '1 commit'. The commit table lists two files: 'app.py' and 'newfile.py', both from 'My first commit' and '6 minutes ago'. At the bottom, there's a light blue banner encouraging the user to 'Add a README' to help people understand the project.

GitHub repository page for `testesGIT` (Public).

Navigation: Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, Settings.

Branches: master (1 branch), 0 tags.

Buttons: Go to file, Add file, Code.

Commit history:

Commit Hash	Author	Message	Time	Commits
0a83266	miNombreDeUsuario	My first commit	6 minutes ago	1 commit

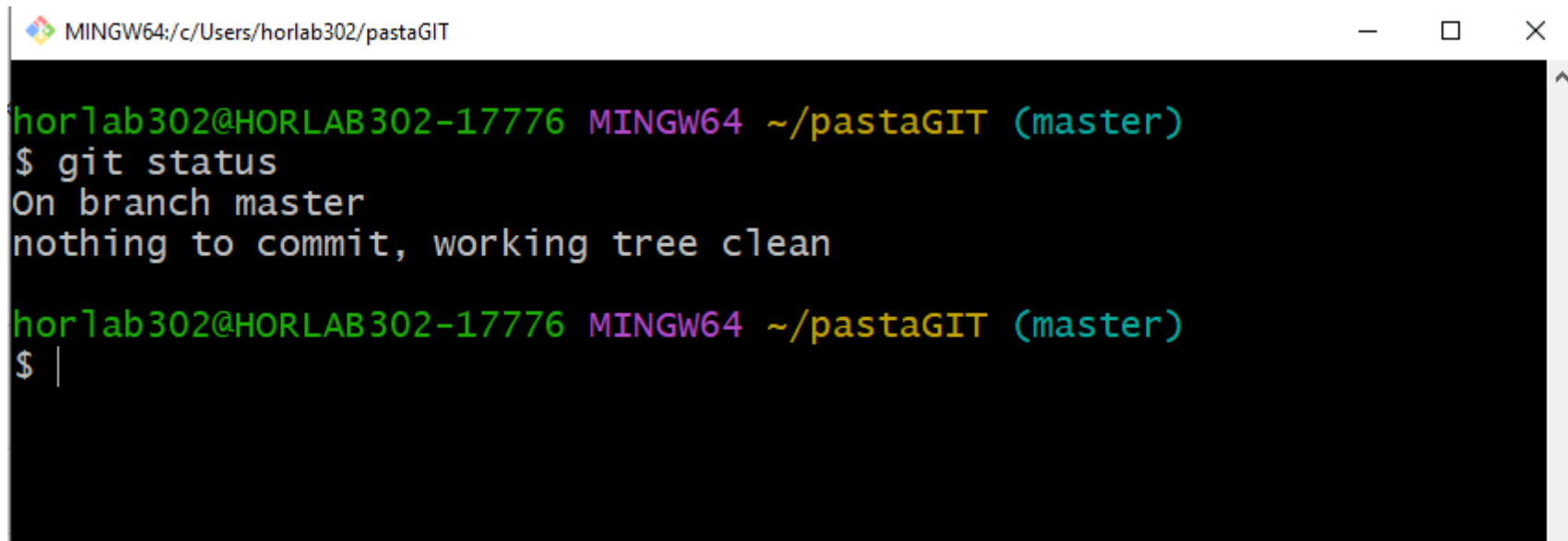
Files in commit:

File	Commit	Time
app.py	My first commit	6 minutes ago
newfile.py	My first commit	6 minutes ago

Help people interested in this repository understand your project by adding a README. [Add a README](#)

Git status

- O git push origin main/master envia os arquivos para o repositório remoto do GIT

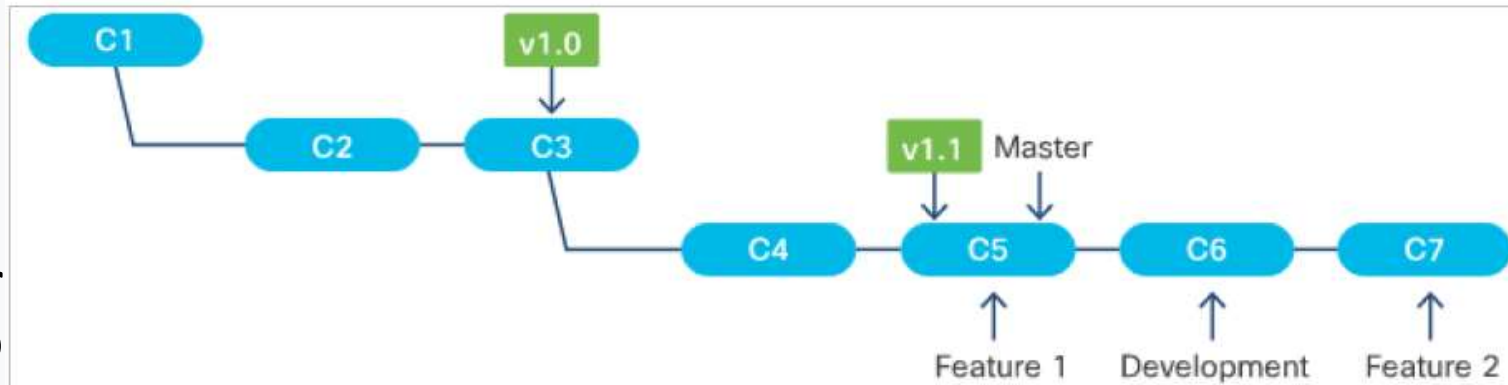
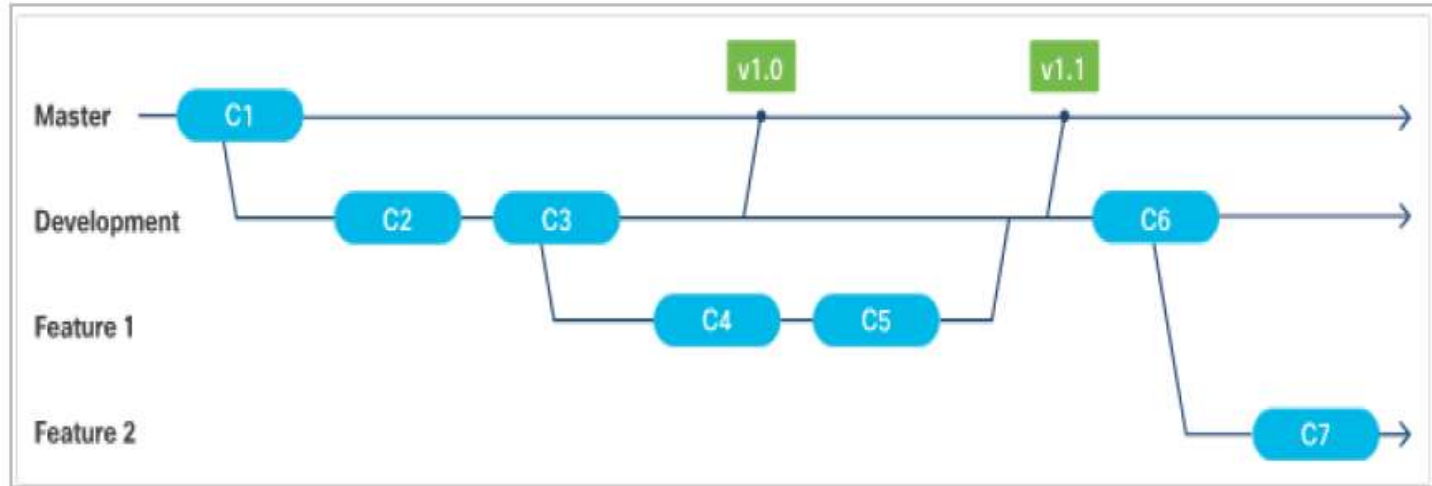


```
MINGW64:/c:/Users/horlab302/pastaGIT
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git status
On branch master
nothing to commit, working tree clean

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ |
```

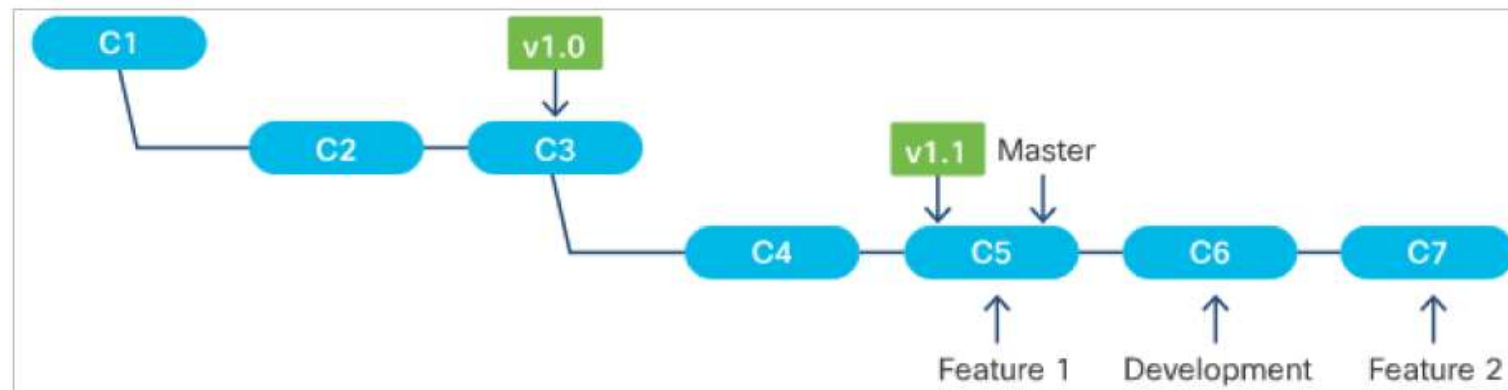
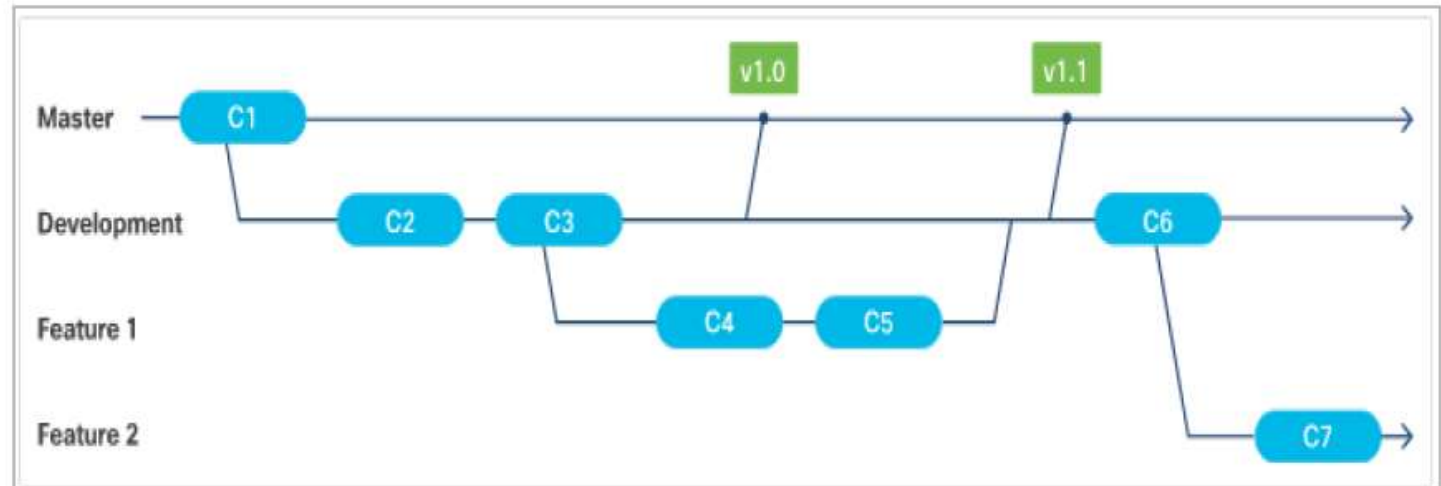
Branch

- Uma branch permite que os usuários trabalhem no código de forma independente sem afetar o código principal no repositório.
- Quando um repositório é criado, o código é colocado automaticamente em uma ramificação chamada Master.
- As branches podem ser locais ou remotas, elas podem ser excluídos e ter seu próprio histórico e diretório de trabalho.



Branch

- A criação de ramificações do Git é leve, e alternar entre ramos é quase instantânea.
- Quando um usuário sai de uma branch para outra, o código em seu diretório local e os arquivos no staging a área mudam, mas o os diretórios do repositório(.git) permanecem inalterado.



branch

- Git checkout -b nome_da_branch
- Git checkout: para mudar de branch
- Git branch: para listar as branches

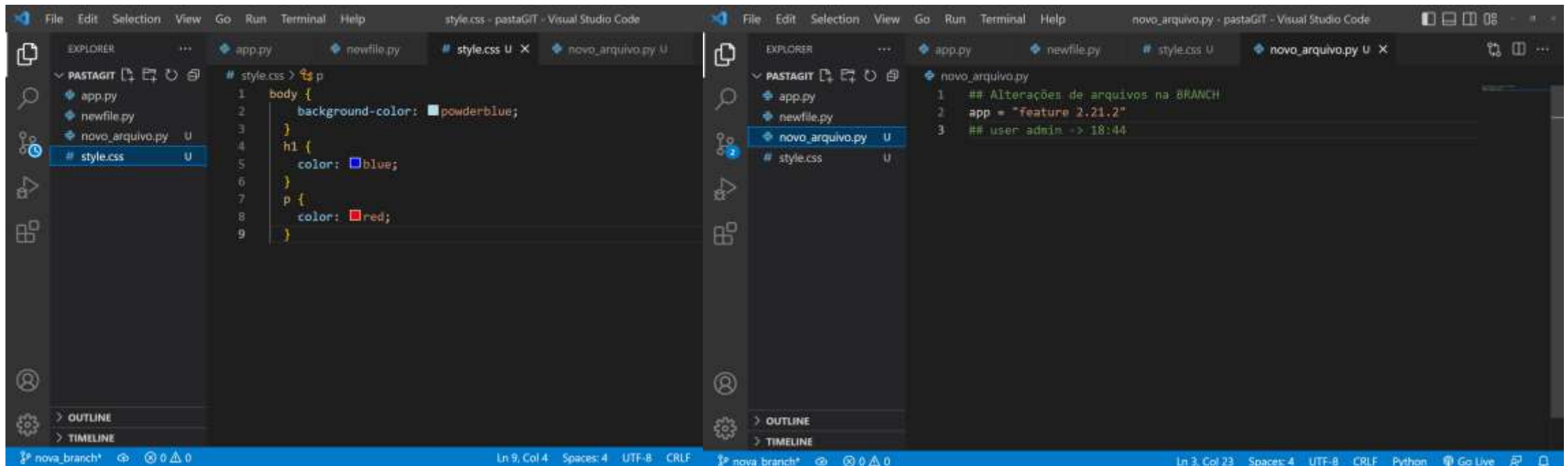
```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git checkout -b nova_branch
Switched to a new branch 'nova_branch'

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git branch
  master
* nova_branch

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ |
```

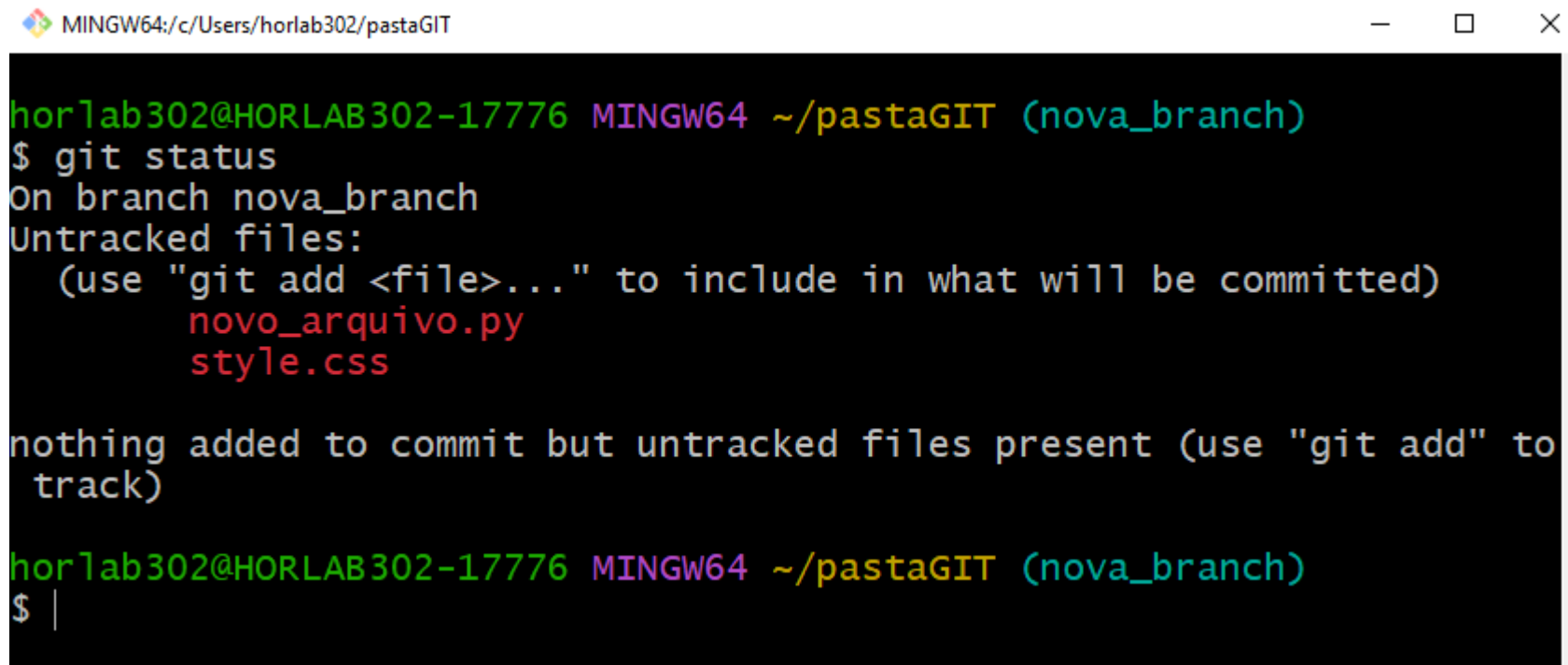
branch

- Criando e editando os arquivos na nova branch



Git status

- Criando e editando os arquivos na nova branch

A screenshot of a Windows terminal window with a black background and colored text. The window title is 'MINGW64:/c:/Users/horlab302/pastaGIT'. The prompt is 'horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)'. The user has entered '\$ git status'. The output shows 'On branch nova_branch', 'Untracked files:', and a list of two files: 'novo_arquivo.py' and 'style.css' in red. A message follows: 'nothing added to commit but untracked files present (use "git add" to track)'. The prompt is repeated, and the user has entered '\$ |' with a cursor.

```
MINGW64:/c:/Users/horlab302/pastaGIT
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git status
On branch nova_branch
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        novo_arquivo.py
        style.css

nothing added to commit but untracked files present (use "git add" to track)
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ |
```

git commit on branch

- Criando e editando os arquivos na nova branch

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git add novo_arquivo.py style.css

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git commit -a -m "enviando novos arquivos da branch"
[nova_branch 473e1fd] enviando novos arquivos da branch
2 files changed, 12 insertions(+)
create mode 100644 novo_arquivo.py
create mode 100644 style.css

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$
```

git push origin branch

- Criando e editando os arquivos na nova branch

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git push origin nova_branch
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 507 bytes | 507.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'nova_branch' on GitHub by visiting
:
remote:      https://github.com/thiagocghc/testesGIT/pull/new/nova_br
anch
remote:
To github.com:thiagocghc/testesGIT.git
 * [new branch]      nova_branch -> nova_branch

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ |
```

git push origin branch

- Criando e editando os arquivos na nova branch

The screenshot shows the GitHub repository page for `thiagocghc/testesGIT`. The repository is public. The navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. The main content area shows the 'Overview' tab selected, with tabs for 'Yours', 'Active', 'Stale', and 'All branches'. A 'New branch' button is visible. The 'Default branch' section shows the `master` branch, updated 20 minutes ago by `miNombreDeUsuario`. A warning message states 'Your master branch isn't protected' and provides a 'Protect this branch' button. The 'Your branches' section shows the `nova_branch` branch, updated 2 minutes ago by `miNombreDeUsuario`, with a 'New pull request' button. The 'Active branches' section also shows the `nova_branch` branch with a 'New pull request' button.

thiagocghc / testesGIT Public

<> Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Search branches...

Overview Yours Active Stale All branches New branch

Default branch

master Updated 20 minutes ago by miNombreDeUsuario Default

Your master branch isn't protected
Protect this branch from force pushing or deletion, or require status checks before merging. [Learn more](#) Dismiss Protect this branch

Your branches


nova_branch Updated 2 minutes ago by miNombreDeUsuario 0 | 1 New pull request




Active branches

nova_branch Updated 2 minutes ago by miNombreDeUsuario 0 | 1 New pull request


git push origin branch





- Criando e editando os arquivos na nova branch

 nova_branch had recent pushes 1 minute ago [Compare & pull request](#)

 nova_branch ▾  2 branches  0 tags [Go to file](#) [Add file ▾](#) [<> Code ▾](#)

This branch is [1 commit ahead](#) of master. [Contribute ▾](#)

 miNombreDeUsuario enviando novos arquivos da branch 473e1fd 3 minutes ago ⌚ 2 commits

 app.py	My first commit	21 minutes ago
 newfile.py	My first commit	21 minutes ago
 novo_arquivo.py	enviando novos arquivos da branch	3 minutes ago
 style.css	enviando novos arquivos da branch	3 minutes ago

Help people interested in this repository understand your project by adding a README. [Add a README](#)

git checkout master

- Criando e editando os arquivos na nova branch

A screenshot of a Windows terminal window with a black background and colored text. The window title is 'MINGW64:/c/Users/horlab302/pastaGIT'. The prompt is 'horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)'. The first command is '\$ git branch', which lists 'master' and '* nova_branch'. The second command is '\$ git checkout master', which outputs 'Switched to branch 'master''. The third command is '\$', with a cursor following it. The prompt now shows '(master)' instead of '(nova_branch)'.

```
MINGW64:/c/Users/horlab302/pastaGIT


horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git branch
  master
* nova_branch


horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git checkout master
Switched to branch 'master'


horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ |
```



git merge

- Criando e editando os arquivos na nova branch

 nova_branch had recent pushes 7 minutes ago [Compare & pull request](#)

 master ▾


 2 branches

 0 tags

[Go to file](#)



[Add file ▾](#)



[Code ▾](#)

 **Your master branch isn't protected**

Protect this branch from force pushing or deletion, or require status checks before merging. [Learn more](#)

[Protect this branch](#) ×

 **miNombreDeUsuario** My first commit 0a83266 26 minutes ago  1 commit

 app.py	My first commit	26 minutes ago
 newfile.py	My first commit	26 minutes ago

Help people interested in this repository understand your project by adding a README.

[Add a README](#)

git merge

- Criando e editando os arquivos na nova branch

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (nova_branch)
$ git checkout master
Switched to branch 'master'

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git merge nova_branch
Updating 0a83266..473e1fd
Fast-forward
 novo_arquivo.py | 3 +++
 style.css       | 9 ++++++++
 2 files changed, 12 insertions(+)
 create mode 100644 novo_arquivo.py
 create mode 100644 style.css

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$
```

git push origin master

- Criando e editando os arquivos na nova branch

```
horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git merge nova_branch
Updating 0a83266..473e1fd
Fast-forward
 novo_arquivo.py | 3 +++
 style.css       | 9 ++++++++
 2 files changed, 12 insertions(+)
 create mode 100644 novo_arquivo.py
 create mode 100644 style.css

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$ git push origin master
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:thiagocghc/testesGIT.git
 0a83266..473e1fd  master -> master

horlab302@HORLAB302-17776 MINGW64 ~/pastaGIT (master)
$
```

git merge

- Repositório master/main com as alterações feitas na branch

The screenshot shows the GitHub interface for a repository named 'testesGIT'. The repository is public. The main navigation bar includes links for Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below the navigation bar, the repository name 'testesGIT' is displayed with a 'Public' badge. A red box highlights the 'master' branch selector and the '2 branches' indicator. Below this, a warning message states 'Your master branch isn't protected' with a 'Protect this branch' button. The commit history shows a recent commit by 'miNombreDeUsuario' titled 'enviando novos arquivos da branch' with a commit hash of '473e1fd' and a timestamp of '10 minutes ago'. The commit details show four files: 'app.py', 'newfile.py', 'novo_arquivo.py', and 'style.css', all with commit messages 'My first commit' or 'enviando novos arquivos da branch'. A green box highlights the commit details section. At the bottom, there is a light blue banner encouraging the user to 'Add a README'.

hc / testesGIT Public

Issues Pull requests Actions Projects Wiki Security Insights Settings

master 2 branches 0 tags Go to file Add file <> Code

Your master branch isn't protected
Protect this branch from force pushing or deletion, or require status checks before merging. [Learn more](#) [Protect this branch](#)

miNombreDeUsuario enviando novos arquivos da branch 473e1fd 10 minutes ago 2 commits

app.py	My first commit	29 minutes ago
newfile.py	My first commit	29 minutes ago
novo_arquivo.py	enviando novos arquivos da branch	10 minutes ago
style.css	enviando novos arquivos da branch	10 minutes ago

Help people interested in this repository understand your project by adding a README. [Add a README](#)