

Introdução a Git e ao GitHub

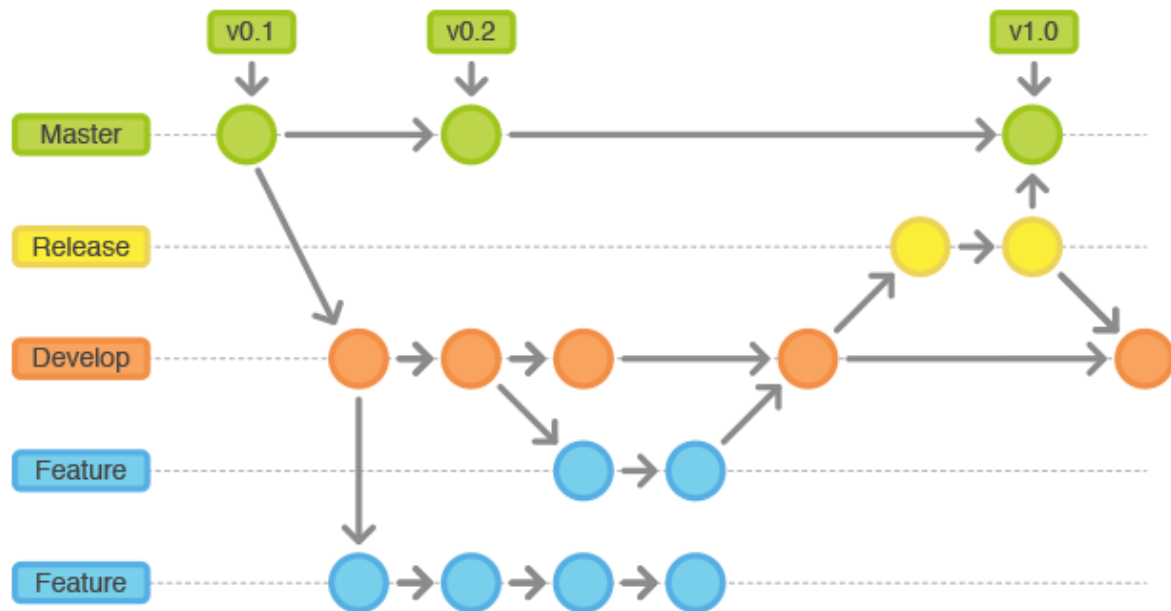
MATEUS SANTOS VALENÇA

Introdução

O que é Git?

O Git é um sistema de controle de versão de projetos. Onde é possível várias pessoas contribuírem de forma que não seja necessário a modificação definitiva do projeto original.

No Git seus projetos ficam salvos localmente no computador.



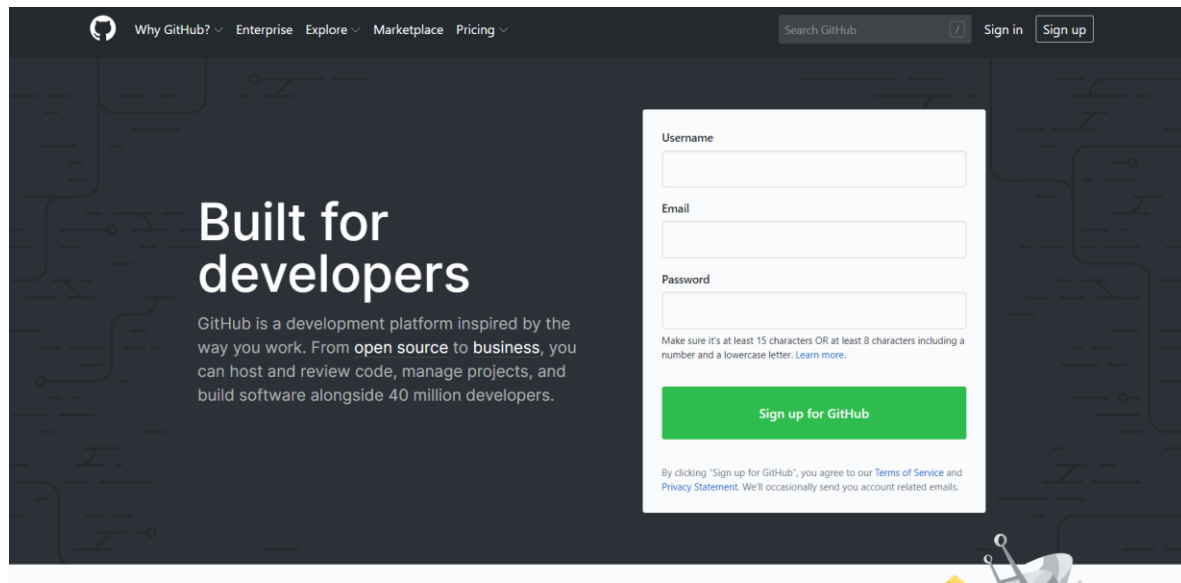
Introdução

- **O que é GitHub?**
 - Ferramenta com Git
 - Gratuito
 - Backup na nuvem dos seus projetos
 - Compartilhamento prático

Primeiros Passos

Criar uma conta

Acessar github.com e criar uma conta



The screenshot shows the GitHub homepage with a dark background and a white sign-up form on the right. The form includes fields for Username, Email, and Password, a green 'Sign up for GitHub' button, and a disclaimer at the bottom. The background features a faint, stylized circuit board pattern.

Why GitHub? ▾ Enterprise ▾ Explore ▾ Marketplace ▾ Pricing ▾

Search GitHub 🔍 Sign in Sign up

Built for developers

GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside 40 million developers.

Username

Email

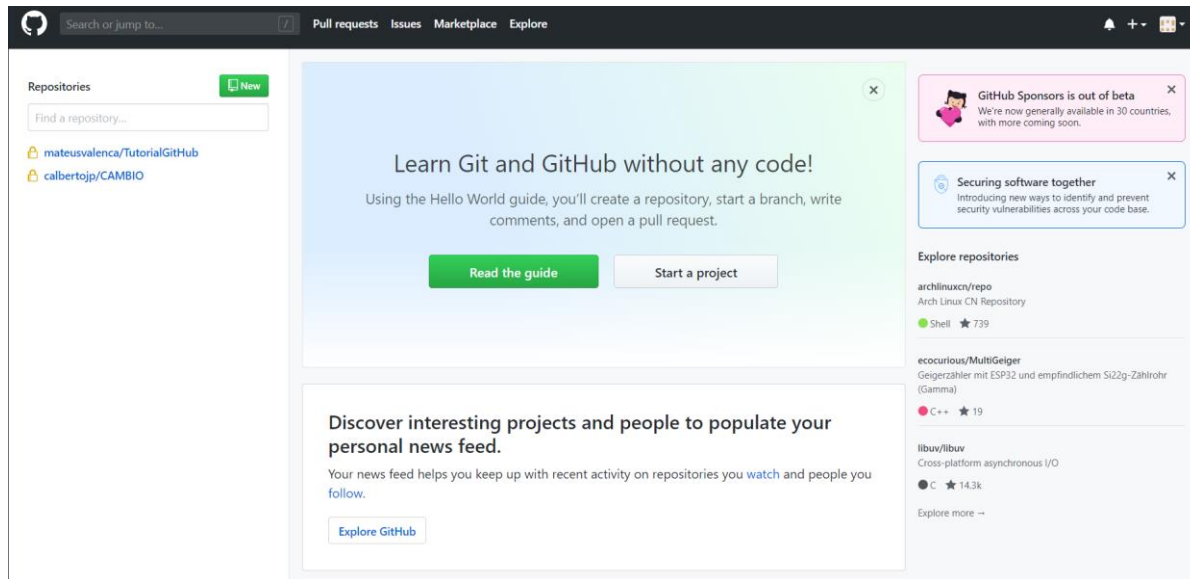
Password

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.

Tela Inicial



Instalado o Git Bash


Para usuários do Windows:

Acessar <https://git-scm.com/download/win> para baixar o git bash

Para usuários do Linux:

No prompt de comando digite:
“sudo apt-get install git”

Git is a member of Software Freedom Conservancy, which handles legal and financial needs for the project. Conservancy is currently raising funds to continue their mission. Consider becoming a supporter!


 **git** --local-branching-on-the-cheap

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.

Downloading Git

 You are downloading the latest (2.25.0) 64-bit version of **Git for Windows**. This is the most recent **maintained build**. It was released **14 days ago**, on 2020-01-13.

[Click here to download manually](#)

Other Git for Windows downloads

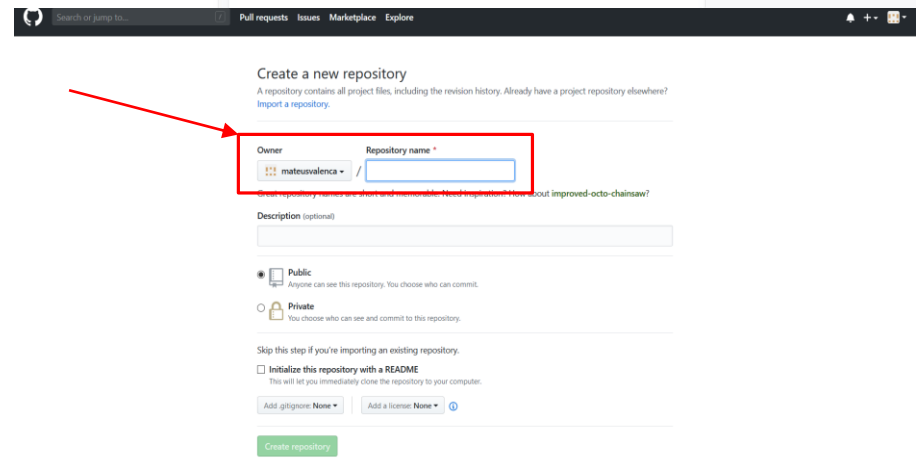
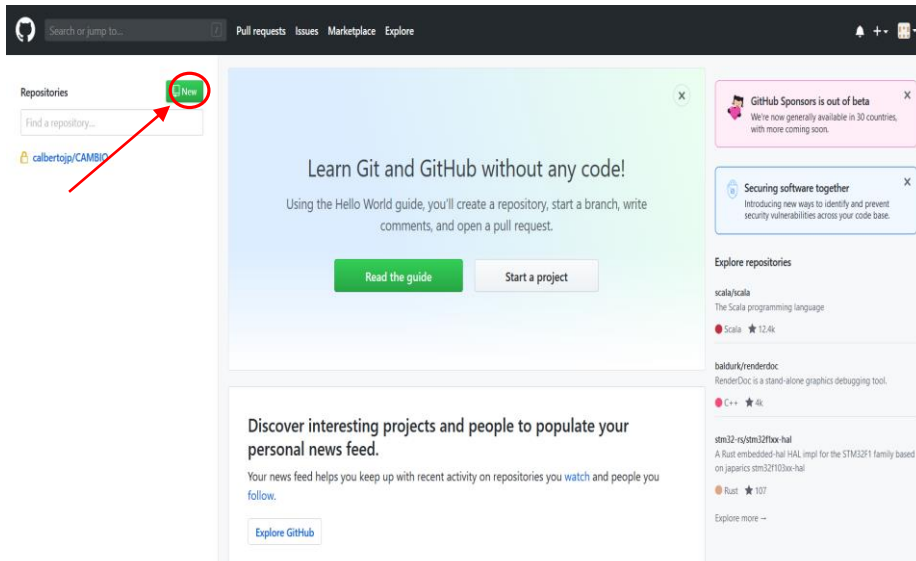
- Git for Windows Setup**
- 32-bit Git for Windows Setup.**
- 64-bit Git for Windows Setup.**
- Git for Windows Portable ("thumbdrive edition")**
- 32-bit Git for Windows Portable.**
- 64-bit Git for Windows Portable.**

The current source code release is version 2.25.0. If you want the *newer* version, you can build it from the **source code**.

Now What?

Now that you have downloaded Git, it's time to start using it.

Criando meu primeiro projeto



Salvando o seu usuário

Com o git bash aberto digite:

git init

git config --global user.name "nome de usuário"

git config --global user.email "e-mail cadastrado"

Essas configurações estarão salvas no seu pc no endereço ~/.gitconfig , onde ~ representa sua home, no meu caso:

C:\Users\mateu\.gitconfig

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho
$ git init
Initialized empty Git repository in C:/Users/mateu/OneDrive/Área de Trabalho/tes
tegit/.git/

mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho (master)
$ git config --global user.name "mateusvalenca"

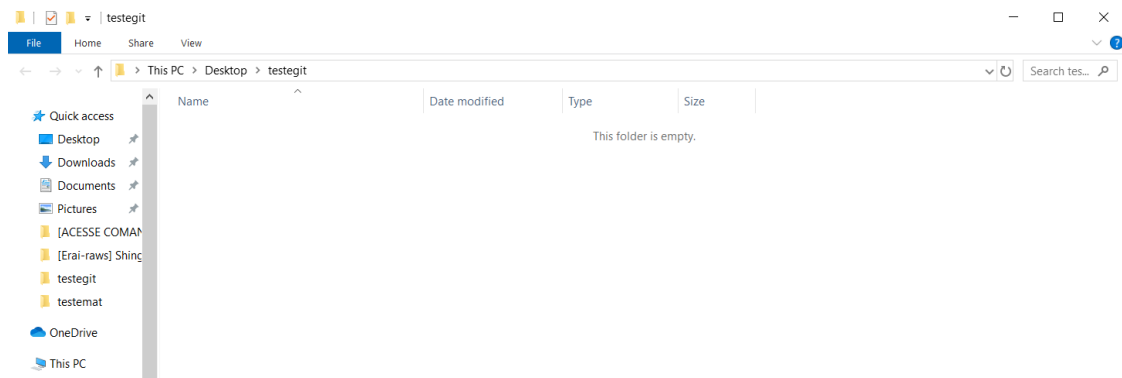
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho (master)
$ git config --global user.email "mateus.valenca@cear.ufpb.br"
```


Clonando o projeto do GitHub

O processo de clonagem permite pegar aquele projeto existente no GitHub e trazer para o nosso git local dentro de uma pasta com aquele nome. O comando usado é:
git clone link_do_projeto

Dica: no bash do Windows para colar algo copiado você usa o botão shift+Insert

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho (master)
$ git clone https://github.com/mateusvalenca/testegit
Cloning into 'testegit'...
warning: You appear to have cloned an empty repository.
```

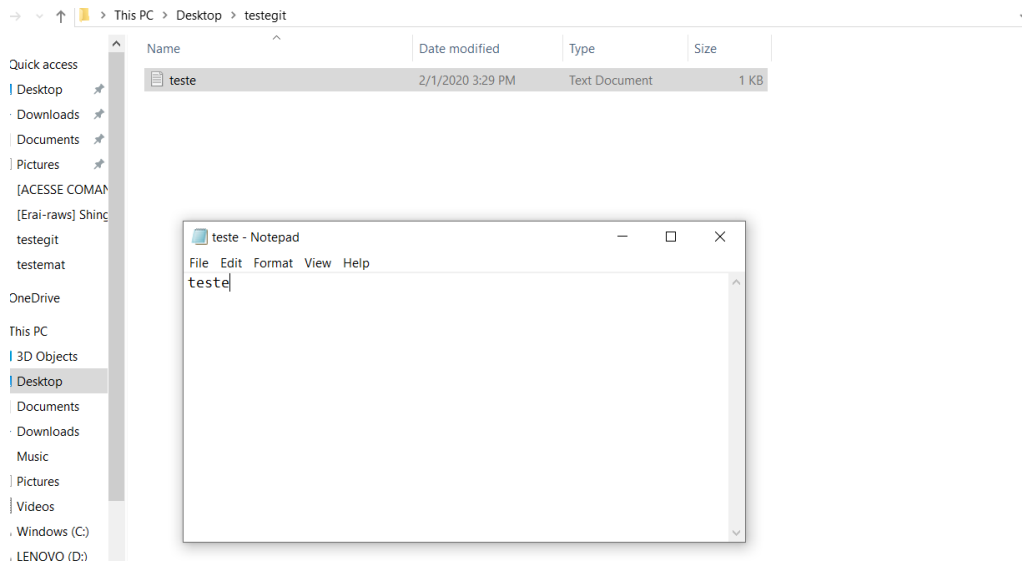


Comandos Iniciais

Git add nome_dos_arquivos ou git add . :

Esse comando vai pegar os arquivos e jogar num lugar chamado INDEX onde ele está sendo preparado para entrar no GitHub na próxima revisão feita.

Crie um arquivo .txt novo e digite algo nele.



Comandos Iniciais

Git commit -m "Mensagem"

Esse comando vai comentar todos os arquivos no INDEX com a mensagem escolhida.

Git push:

Insere os arquivos do INDEX no seu GitHub

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git add .

mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git commit -m "adição arquivo teste"
[master (root-commit) 39605c1] adição arquivo teste
 1 file changed, 1 insertion(+)
 create mode 100644 teste.txt

mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git push
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 245 bytes | 81.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/mateusvalenca/testegit
 * [new branch]      master -> master
```

Mudanças feitas

The screenshot shows the GitHub interface for a repository named 'testegit' by user 'mateusvalencia'. The repository is marked as 'Private'. At the top, there are buttons for 'Unwatch' (1), 'Star' (0), and 'Fork' (0). Below this is a navigation bar with links to 'Code', 'Issues' (0), 'Pull requests' (0), 'Actions', 'Projects' (0), 'Security', 'Insights', and 'Settings'. The main content area shows a message: 'No description, website, or topics provided.' with an 'Edit' button. Below this is a 'Manage topics' link. A summary bar displays '1 commit', '1 branch', '0 packages', and '0 releases'. Below the summary bar are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and a green 'Clone or download' button. The commit history shows one commit by 'mateusvalencia' titled 'adição arquivo teste', with the latest commit '39685c1' made '2 minutes ago'. A file named 'teste.txt' is listed as part of this commit. At the bottom, there is a light blue box with the text 'Add a README with an overview of your project.' and a green 'Add a README' button.

mateusvalencia / testegit Private

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Security Insights Settings

No description, website, or topics provided. Edit

Manage topics

1 commit 1 branch 0 packages 0 releases

Branch: master New pull request Create new file Upload files Find file Clone or download

"mateusvalencia" ·adição arquivo teste Latest commit 39685c1 2 minutes ago

teste.txt ·adição arquivo teste 2 minutes ago

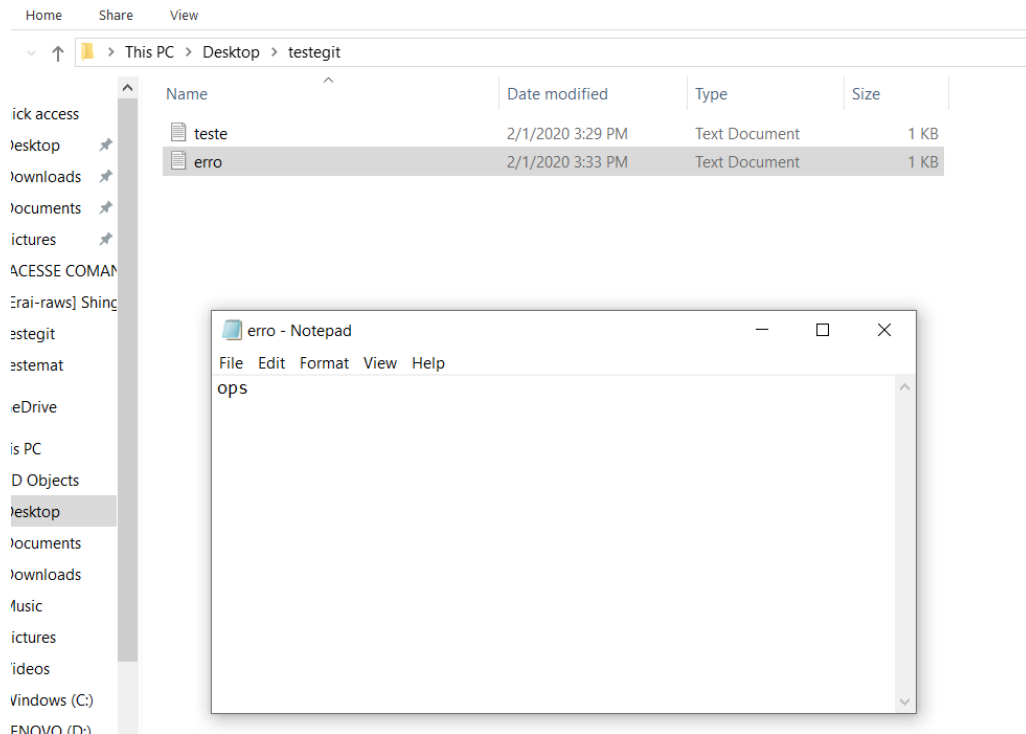
Add a README with an overview of your project. Add a README

E se a alteração está errada?

No seu git, faça um novo arquivo e repita o processo anterior para adicioná-lo ao seu git:

Dica:

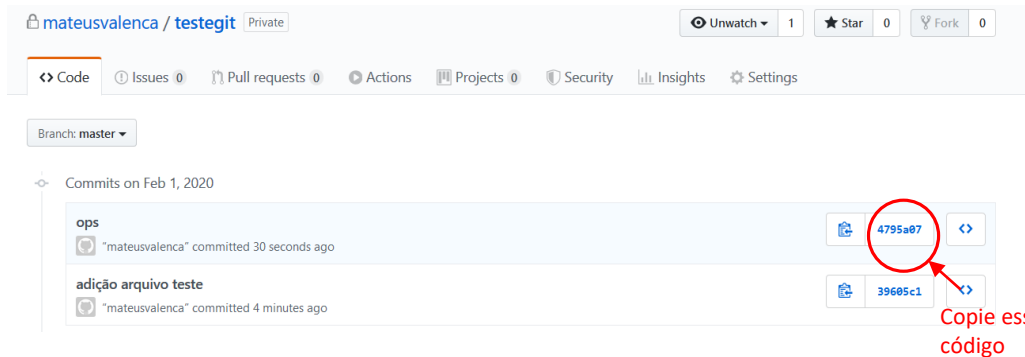
- 1- Git add .
- 2- git commit -m "mensagem"
- 3- git push link



Revertendo o que foi feito!

No GitHub, vá na aba commits e veja o último commit feito.

Comando: `git revert <código>`



```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git revert 4795a07
```

Revertendo o que foi feito!

Nesta tela você tem opção de inserir comentários, aperte qualquer tecla e em seguida vá para a linha onde deseja comentar. Quando terminar aperte esc para sair do modo de edição e digite “:qa!” sem aspas e enter para finalizar.

```
MINGW64:/c/Users/mateu/OneDrive/Área de Trabalho/testegit
```

```
Revert "ops"
```

```
This reverts commit 4795a0728f2a592398fd1631e8f053593b47d860.
```

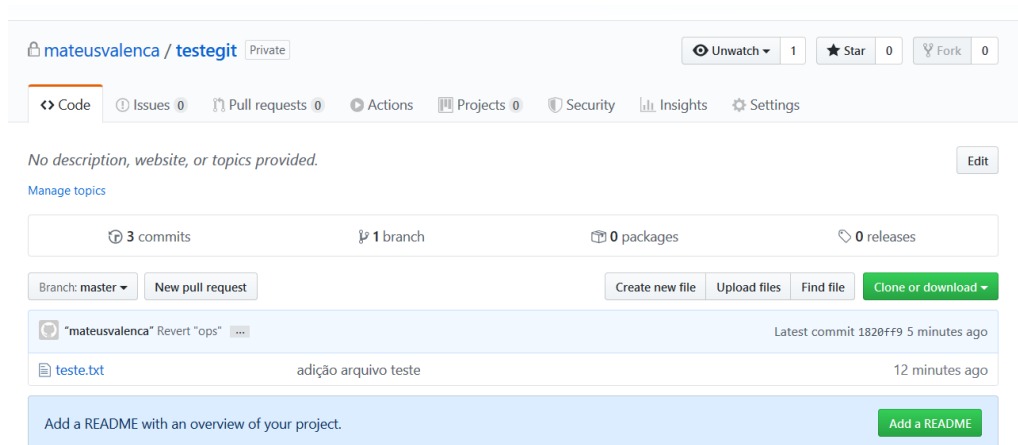
```
# Please enter the commit message for your changes. Lines starting  
# with '#' will be ignored, and an empty message aborts the commit.  
#  
# On branch master  
# Your branch is up to date with 'origin/master'.  
#  
# Changes to be committed:  
#   deleted:    erro.txt  
#
```

```
Revisão de Erro
```

```
< Trabalho/testegit/.git/COMMIT_EDITMSG[+] [unix] (15:36 01/02/2020)14,16-15 All  
:qa!
```

Revertendo o que foi feito!

A reversão foi local, para mudar no GitHub basta usar git push.



Branch

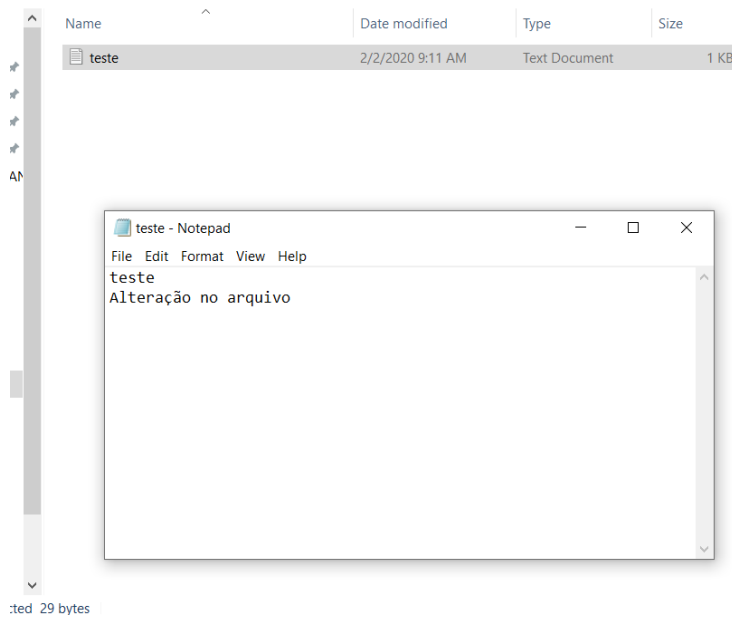
Para começar precisamos criar o branch na pasta local.
O branch criado vai ser uma cópia do master, mas o que for alterado aqui não é alterado no master.

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git branch novo

mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git checkout novo
Switched to branch 'novo'
```

Branch

Vamos praticar!
Altere o arquivo Testeadd.txt e em seguida tente fazer o de sempre:
git add .
git commit -m "mensagem"
git push



Indicando quem é o master do seu branch

O comando

`git --set-upstream origin branch`

Só é necessário usar na primeira vez que utilizar o push nesse branch, depois só:

`git push`

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (novo)
$ git push
fatal: The current branch novo has no upstream branch.
To push the current branch and set the remote as upstream, use

    git push --set-upstream origin novo


mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (novo)
$ git push --set-upstream origin novo
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Writing objects: 100% (3/3), 305 bytes | 101.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
remote:
remote: Create a pull request for 'novo' on GitHub by visiting:
remote:   https://github.com/mateusvalenca/testegit/pull/new/novo
remote:
To https://github.com/mateusvalenca/testegit
 * [new branch]      novo -> novo
Branch 'novo' set up to track remote branch 'novo' from 'origin'.
```

The screenshot shows the GitHub repository page for 'testegit'. At the top, there are statistics: 3 commits, 2 branches, 0 packages, and 0 releases. Below this, a yellow banner indicates 'Your recently pushed branches:' with a branch named 'novo' pushed 1 minute ago. A 'Compare & pull request' button is visible. The main content area shows a list of branches. The 'master' branch is selected and marked as 'default'. The 'novo' branch is also listed. A dropdown menu is open, showing the 'Switch branches/tags' section with a search bar and a list of branches including 'master' and 'novo'. Other buttons like 'Create new file', 'Upload files', 'Find file', 'Clone or download', and 'Add a README' are also visible.

Requisitando a mudança no master


4 commits 2 branches 0 packages 0 releases


Your recently pushed branches:

 **novo** (3 minutes ago) Compare & pull request

Branch: **novo** ▾ **New pull request** [Create new file](#) [Upload files](#) [Find file](#) [Clone or download ▾](#)

This branch is 1 commit ahead of master. [Pull request](#) [Compare](#)



 **"mateusvalenca"** Alteração no arquivo teste Latest commit 45b7cd1 4 minutes ago


 **teste.txt** Alteração no arquivo teste 4 minutes ago










Add a README with an overview of your project. [Add a README](#)

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](#).

 base: **master** ▾  compare: **novo** ▾ ✓ Able to merge. These branches can be automatically merged.

 Alteração no arquivo teste

Write Preview **AA** **B** *i*       @   

Foi adicionada uma linha de código

Attach files by dragging & dropping, selecting or pasting them. 

[Create pull request](#)

Master verificando mudanças

Primeiro volte ao ramo master:
git checkout master

Cheque se há diferenças entre os
ramos:
git diff branch

Verificado as diferenças, estou de
acordo e quero mudar o master:
git merge branch

Isso vai mesclar os branches
localmente, depois é necessário
usar git push para realizar as
alterações no GitHub

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (novo)
$ git checkout master
Switched to branch 'master'
Your branch is up to date with 'origin/master'.

mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git diff novo
diff --git a/teste.txt b/teste.txt
index 0d16cee..bb8abfc 100644
--- a/teste.txt
+++ b/teste.txt
@@ -1,2 +1 @@
-teste
-Alteração no arquivo
\ No newline at end of file
+teste
\ No newline at end of file
```

```
mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git merge novo
Updating 1820ff9..45b7cd1
Fast-forward
 teste.txt | 3 ++-
 1 file changed, 2 insertions(+), 1 deletion(-)

mateu@LAPTOP-14FT9S2U MINGW64 ~/OneDrive/Área de Trabalho/testegit (master)
$ git push
Total 0 (delta 0), reused 0 (delta 0)
To https://github.com/mateusvalenca/testegit
 1820ff9..45b7cd1 master -> master
```

Vamos Praticar!

Primeiro acessem:

<https://github.com/mateusvalenca/matriculas>

E clonem esse projeto.