



Controle de Versão

Por Marcos Taborda

Controle de versão?

É um sistema onde registra as mudanças feitas em um ou mais arquivo(s), possibilitando a recuperação dele(s) se necessário.

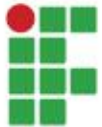


Motivos para versionar seu código?

Organização, possibilita voltar para versões anteriores do código.

Segurança, não perde código, pois tem um histórico dele.

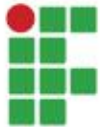
Aprender a trabalhar em equipe, dois desenvolvedores podem alterar o mesmo arquivo ao mesmo tempo no projeto.



O que é git?

O git é um controle de versão criado em 2005 por Linus Torvalds e foi utilizado no gerenciamento de código do kernel Linux.

Ele é Open Source e muito utilizado por desenvolvedores de Software Livre, e não dependente de acesso a uma rede ou a um servidor central.

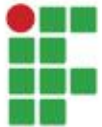


Instalação do Git

A instalação do git para Windows, Linux ou iOS, está disponível neste link

<https://git-scm.com/downloads>

Agora você deve acessar o terminal do Linux ou iOS, e no Windows acessar o Terminal do Windows.



Git - Instalação do Git

Verifique se o git está instalado, comando:

git

```
C:\WINDOWS\system32\cmd.exe
```

```
Microsoft Windows [versão 10.0.16299.64]
```

```
(c) 2017 Microsoft Corporation. Todos os direitos reservados.
```

```
C:\Users\marco>git
```

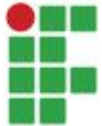
```
usage: git [--version] [--help] [-C <path>] [-c name=value]
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
        [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
        <command> [<args>]
```

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)

clone	Clone a repository into a new directory
-------	---

init	Create an empty Git repository or reinitialize an existing one
------	--



Git - Configuração

C:\WINDOWS\system32\cmd.exe

```
C:\Users\marco>git config --global user.name "Marcos Taborda"

C:\Users\marco>git config --global user.email "marcos.tabordamail@gmail.com"

C:\Users\marco>git config --global color.ui true

C:\Users\marco>
```

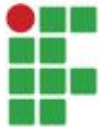
Configurar nome e e-mail:

```
git config --global user.name "seu nome"
```

```
git config --global user.email "email@gmail.com"
```

Para colorir as interações do git:

```
git config --global color.ui true
```



INSTITUTO FEDERAL
Rio Grande do Sul
Campus Porto Alegre



GitHub

É um serviço online para fornecer **repositório para o Git na Web**, possibilita compartilhar o código com outras pessoas e também colaborar com a comunidade de desenvolvedores e participar de projetos Open Source.

Ele é amplamente utilizado, pois concede uma conta gratuita para qualquer iniciativa Open Source, ou seja, você pode criar ilimitados repositórios no GitHub, hoje em dia ele é uma comunidade com **3 milhões de usuários**.


Com essa possibilidade de criar repositórios, se torna uma vitrine para nossos códigos.



GitHub - Criando conta


Acesse <https://github.com/join>


Guarde:
Usuário e senha.


 [Features](#) [Business](#) [Explore](#) [Marketplace](#) [Pricing](#) [Sign in](#) or [Sign up](#)

Join GitHub

The best way to design, build, and ship software.

 **Step 1:**
Create personal account

 **Step 2:**
Choose your plan

 **Step 3:**
Tailor your experience

Create your personal account

Username

This will be your username. You can add the name of your organization later.

Email address

We'll occasionally send updates about your account to this inbox. We'll never share your email address with anyone.

Password

Use at least one lowercase letter, one numeral, and seven characters.

You'll love GitHub

- Unlimited collaborators
- Unlimited public repositories
- ✓ Great communication
- ✓ Frictionless development
- ✓ Open source community

GitHub - Criação do repositório

The screenshot shows the GitHub web interface. At the top, there's a navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. Below this, the user profile 'marcostaborda' is visible. The main content area is divided into two sections: 'Browse activity' and 'Discover repositories'. The 'Discover repositories' section shows a list of recent pushes to various repositories, including 'asaphoto-pequenoastro2017' and 'multas.dpsul.com.br'. On the right side, there are two panels: 'Repositories you contribute to' (12) and 'Your repositories' (37). The 'New repository' button in the 'Your repositories' panel is highlighted with a red box.

Search GitHub Pull requests Issues Marketplace Explore

marcostaborda

Browse activity Discover repositories

jefersoncunha pushed to `master` in jefersoncunha/asaphoto-pequenoastro2017 3 days ago
8429fcb Merge branch 'master' of https://github.com/jefersoncunha/asaph...
3109d98 resultado
View comparison for these 2 commits »

jefersoncunha pushed to `master` in jefersoncunha/multas.dpsul.com.br 4 days ago
82c7483 style hover menu

jefersoncunha pushed to `master` in jefersoncunha/multas.dpsul.com.br 4 days ago
12cea5b link hover cp

jefersoncunha pushed to `master` in jefersoncunha/multas.dpsul.com.br 4 days ago
e3d8df7 botão mapa style

jefersoncunha pushed to `master` in jefersoncunha/multas.dpsul.com.br 4 days ago
22c3d2b footer copyrhgt

jefersoncunha pushed to `master` in jefersoncunha/multas.dpsul.com.br 4 days ago
f4f29f2 copyright link

jefersoncunha pushed to `master` in jefersoncunha/multas.dpsul.com.br 4 days ago
e79f711 estilo mapa route

Repositories you contribute to 12

- jefersoncunha/dpsul 0 ★
- jefersoncunha/simoesepi 1 ★
- jefersoncunha/matematica.com.br 0 ★
- jefersoncunha/multas.dpsul.com.... 0 ★
- jefersoncunha/stangherlinepiefer... 0 ★

Load more...

Your repositories 37

Find a repository...

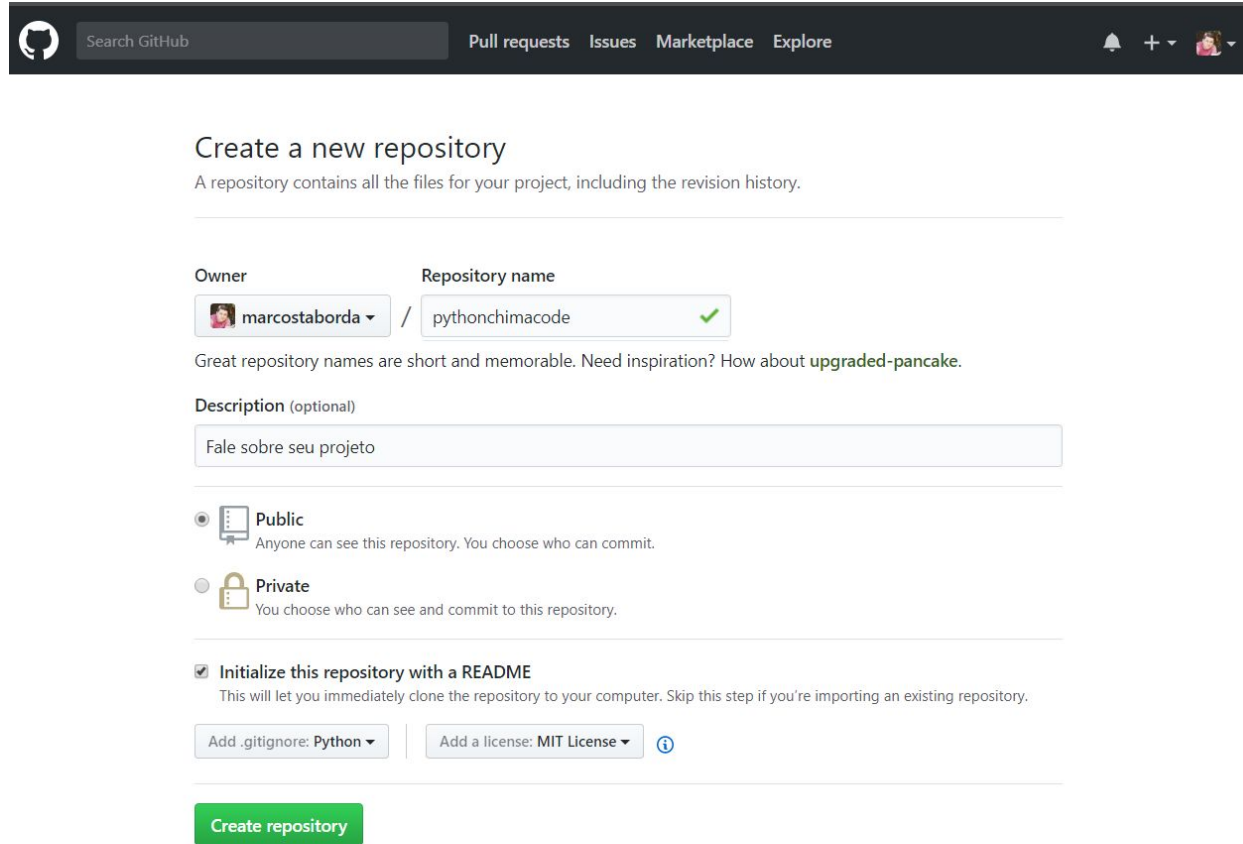
All Public Private Sources Forks

- jefersoncunha/rockgaucho.com
- jefersoncunha/asaphoto-pequenoastro2017
- jefersoncunha/multas.dpsul.com.br
- jefersoncunha/matematica.com.br
- page-chimacode
- jefersoncunha/dpsul
- asynctask

Load more...

Click em
“New repository”

GitHub - Criação do repositório



The screenshot shows the GitHub interface for creating a new repository. At the top is a dark navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. Below this, the main heading is 'Create a new repository' with a subtext: 'A repository contains all the files for your project, including the revision history.' The form includes an 'Owner' dropdown set to 'marcostaborda', a 'Repository name' field with 'pythonchimacode' and a green checkmark, and a 'Description (optional)' text area with the placeholder 'Fale sobre seu projeto'. There are two radio button options: 'Public' (selected) and 'Private'. A checkbox 'Initialize this repository with a README' is checked. At the bottom, there are dropdowns for 'Add .gitignore: Python' and 'Add a license: MIT License', followed by a green 'Create repository' button.

1. Nome Repositório
2. Add gitignore
3. Add licença

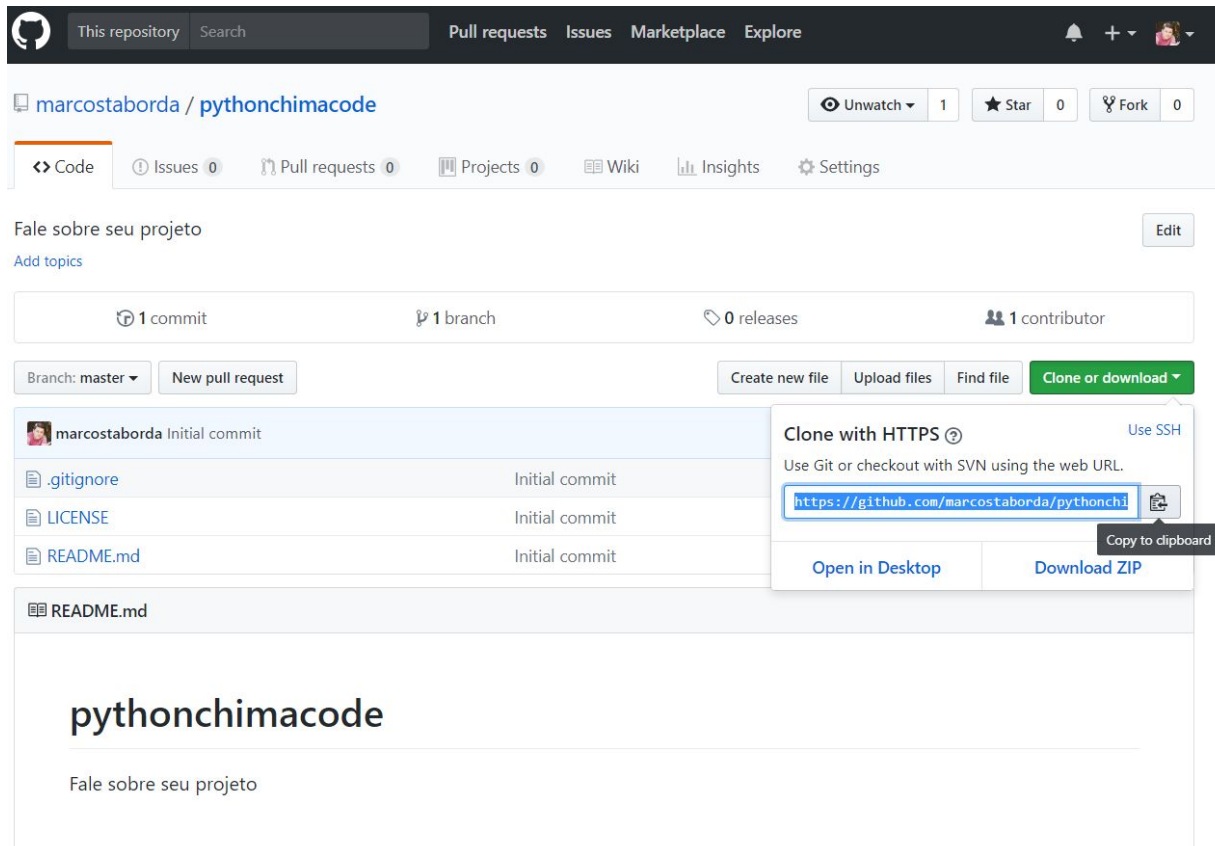
A descrição é opcional

MIT license - o usuário de um trabalho derivado precisa obter a licença tanto do autor deste trabalho como também dos detentores dos direitos de cada componente que faz parte dele.



GitHub - Clonar repositório

1. Copiar o link



The screenshot shows the GitHub interface for the repository 'marcostaborda / pythonchimacode'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The 'Clone or download' button is highlighted, and its dropdown menu is open, showing the option to 'Clone with HTTPS' with the URL 'https://github.com/marcostaborda/pythonchi'. The dropdown menu also includes options for 'Use SSH', 'Open in Desktop', and 'Download ZIP'. A 'Copy to clipboard' tooltip is visible over the URL. The repository files list includes '.gitignore', 'LICENSE', and 'README.md', all marked as 'Initial commit'. The repository name 'pythonchimacode' is displayed prominently, along with a link to 'Fale sobre seu projeto'.

marcostaborda / pythonchimacode

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Fale sobre seu projeto Edit

Add topics

1 commit 1 branch 0 releases 1 contributor

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

marcostaborda Initial commit

File	Status
.gitignore	Initial commit
LICENSE	Initial commit
README.md	Initial commit

README.md

pythonchimacode

Fale sobre seu projeto

Git - Clonar repositório

C:\WINDOWS\system32\cmd.exe

```
C:\Users\marco\Documents\GitChimaCode>git clone https://github.com/marcostaborda/pythonchimacode.git
Cloning into 'pythonchimacode'...
remote: Counting objects: 5, done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), done.
```

COMANDOS	DESCRIÇÃO
<code>git clone <linkGitHubRepositorio></code>	Obtém o repositório remoto
<code>dir</code>	Lista o conteúdo da pasta
<code>cd <pastaClonada></code>	Entra no diretório clonado



Git - Rastreando o Arquivo

1. Criar arquivo no repositório git

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git status
On branch master
Your branch is up-to-date with 'origin/master'.
Untracked files:
  (use "git add <file>..." to include in what will be committed)

    testeaddgit.txt

nothing added to commit but untracked files present (use "git add" to track)
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git add testeaddgit.txt
```

<code>git status</code>	Checar estado do repositório
<code>git add <nome arquivo></code>	Add o arquivo ao rastreamento Git

Git mantém 3 “árvores” de fluxos de trabalho:

A primeira é “Working Directory”, quando é alterado arquivo(s) ou criados arquivos novos dentro do diretório. Esses arquivos não estão versionados(rastreados) no git.

Após o comando add ele vai para o segundo fluxo “Index(Stage)”, que é um “árvore” temporária.



Git - Gravando o Arquivo

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git status
On branch master
Your branch is up-to-date with 'origin/master'.
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

       new file:   testeaddgit.txt
```

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git commit -m "Primeiro commit"
[master 1c34814] Primeiro commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 testeaddgit.txt
```

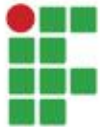
git status

Checar estado do repositório

git commit -m "<mensagem>"

Confirmando alterações.

O terceiro fluxo de trabalho é quando gravamos as alterações. Quando damos um commit, estamos confirmando o arquivo será enviado para a “HEAD” que aponta para o último commit. **Mas ainda não está no repositório remoto.**



INSTITUTO FEDERAL
Rio Grande do Sul
Campus Porto Alegre



Git - Fluxo de Trabalho



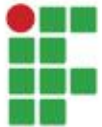
Git Push- Enviando alterações para o GitHub

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git push origin master
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 281 bytes | 0 bytes/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/marcostaborda/pythonchimacode.git
    6c0dac7..1c34814  master -> master
```

git push origin master

Envia as alterações para o repositório no Github


O envio de alterações para o servidor é chamado de:
push



INSTITUTO FEDERAL
Rio Grande do Sul
Campus Porto Alegre



GitHub - Visualizar Alterações

 This repository Search Pull requests Issues Marketplace Explore

marcostaborda / pythonchimacode

Unwatch 1 Star 0 Fork 0


[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#) [Settings](#)





Fale sobre seu projeto [Edit](#)


[Add topics](#)

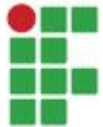
[2 commits](#) [1 branch](#) [0 releases](#) [1 contributor](#) [MIT](#)

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

 **marcostaborda** Primeiro commit Latest commit 1c34814 29 minutes ago

 .gitignore	Initial commit	2 hours ago
 LICENSE	Initial commit	2 hours ago
 README.md	Initial commit	2 hours ago
 testeaddgit.txt	Primeiro commit	29 minutes ago


 [README.md](#)



INSTITUTO FEDERAL
Rio Grande do Sul
Campus Porto Alegre



GitHub - Visualizar Alterações


 This repository Search Pull requests Issues Marketplace Explore


marcostaborda / pythonchimacode Unwatch 1 Star 0 Fork 0


[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#) [Settings](#)

Primeiro commit

[Browse files](#)

 marcostaborda committed 31 minutes ago 1 parent 6c0dac7 commit 1c348144c3332239ee1c4a0b782f87a28ddb4b

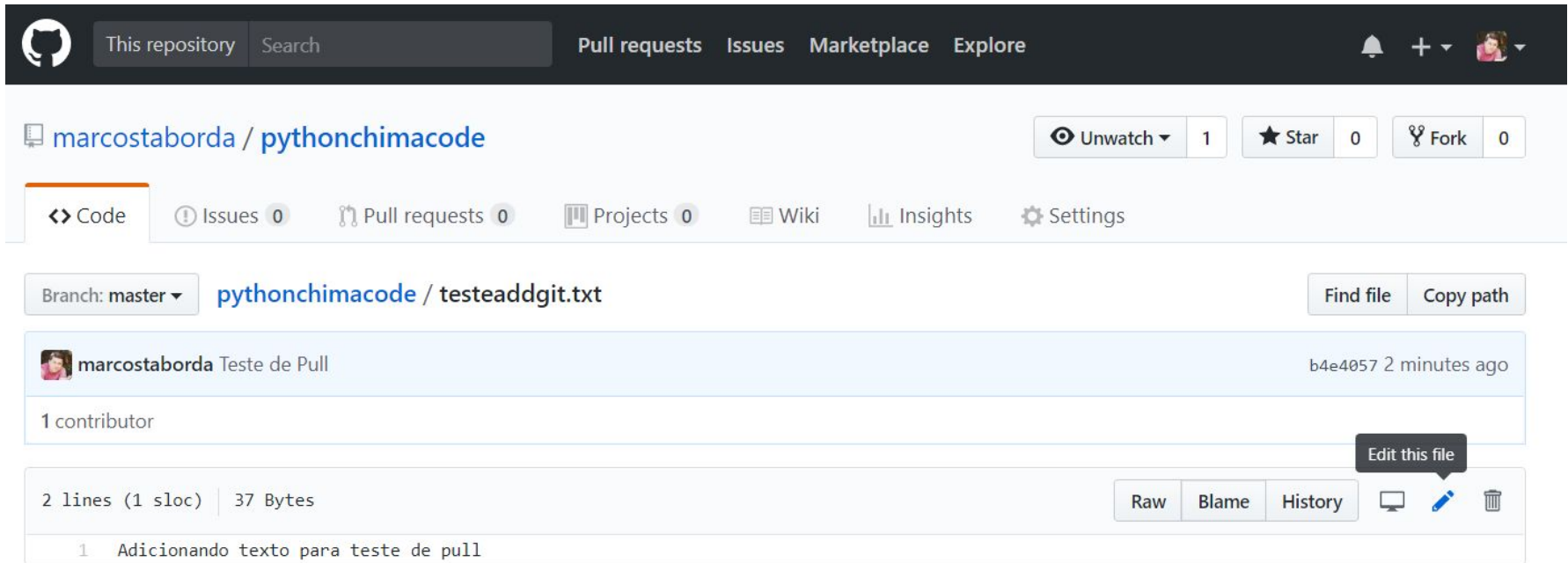
 Showing 1 **changed file** with 0 additions and 0 deletions. Unified Split

0  testeaddgit.txt View

No changes.

Git - Atualizando Repositório Local - Pull

1. **Alterar** arquivo dentro do GitHub.



marcostaborda / pythonchimacode

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Branch: master pythonchimacode / testeaddgit.txt Find file Copy path

marcostaborda Teste de Pull b4e4057 2 minutes ago

1 contributor

2 lines (1 sloc) 37 Bytes Raw Blame History Edit this file

```
1 Adicionando texto para teste de pull
```

git pull

Puxa as alterações do servidor para o repositório local

Git - Histórico

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git log
commit 1c348144c3332239eelc4a0b782f87a28ddb4b (HEAD -> master, origin/master, origin/HEAD)
Author: Marcos Taborda <marcos.tabordamail@gmail.com>
Date: Thu Nov 16 01:38:34 2017 -0200
```

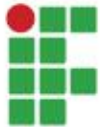
Primeiro commit

```
commit 6c0dac74fd2c8a317e55d815483ad458b16fc523
Author: Marcos Taborda <marcos.tabordamail@gmail.com>
Date: Thu Nov 16 00:10:55 2017 -0200
```

Initial commit

`git log`

Retorna a lista(histórico), de commits com autor e data.



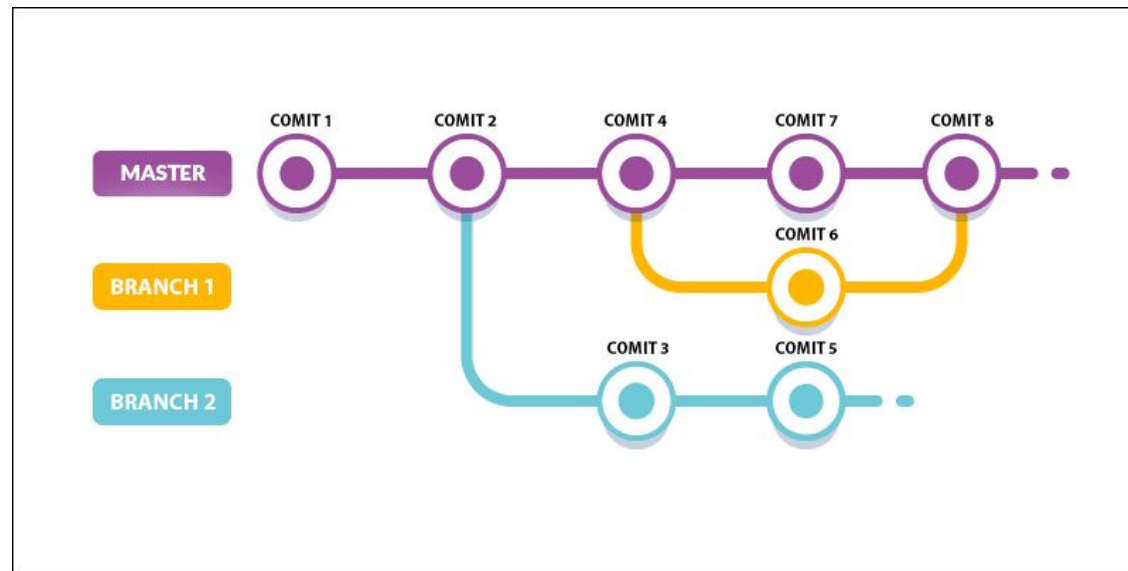
INSTITUTO FEDERAL
Rio Grande do Sul
Campus Porto Alegre



Git – Ramificação (Branch)

São utilizados para separar novas funcionalidades do que está pronto. Imagine que você tem um site no ar, e precisa testar algo novo para o layout, porém não quer que mude o que está no ar.

Para isso criamos uma ramificação(branch) do diretório padrão(master) e quando terminamos mesclam(merge) os códigos.



Git – Criando Ramificação (Branch)

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git branch
* master

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git checkout -b funcionalidade_nova
Switched to a new branch 'funcionalidade_nova'

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git branch
* funcionalidade_nova
  master

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>_
```

<code>git branch</code>	Lista ramificações no repositório local.
<code>git branch -a</code>	Lista ramificações locais e no GitHub(remote)
<code>git checkout -b <nome branch></code>	Cria e já seleciona a nova ramificação
<code>git checkout master</code>	Seleciona a branch master
<code>git checkout <nome branch></code>	Seleciona a branch criada

A branch que está selecionada estará marcada com *



Git – Criando arquivos (Branch)

1. **Crie** um ou mais arquivos no repositório local.

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git status
On branch funcionalidade_nova
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   testeArquivoBranch.txt

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

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git add .

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git commit -m "Arquivo para testar branch"
[funcionalidade_nova 10599d2] Arquivo para testar branch
 1 file changed, 1 insertion(+), 1 deletion(-)

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>
```

O fluxo de trabalho é o mesmo, tem que adicionar a Index e efetuar um commit depois



Git – Enviando a Branch para o GitHub

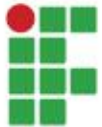
```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git push origin funcionalidade_nova
Counting objects: 6, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (6/6), 598 bytes | 0 bytes/s, done.
Total 6 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/marcostaborda/pythonchimacode.git
 * [new branch]      funcionalidade_nova -> funcionalidade_nova

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git branch -a
* funcionalidade_nova
  master
  remotes/origin/HEAD -> origin/master
  remotes/origin/funcionalidade_nova
  remotes/origin/master

C:\Users\marco\Documents\GitChimaCode\pythonchimacode>
```

`git push origin <nome branch>`

Envia a branch para o GitHub



INSTITUTO FEDERAL
Rio Grande do Sul
Campus Porto Alegre



Git – Mesclando ramificações

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git checkout master
Switched to branch 'master'
Your branch is up-to-date with 'origin/master'.
```

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>git merge funcionalidade_nova
Updating b4e4057..10599d2
Fast-forward
 testeArquivoBranch.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 testeArquivoBranch.txt
```

```
C:\Users\marco\Documents\GitChimaCode\pythonchimacode>_
```

<code>git checkout <branch destino></code>	Seleciona a branch que vai receber as alterações
<code>git merge <branch origem></code>	Mescla as alterações da branch origem com a selecionada(destino).
<code>git push origin master</code>	Enviar a master para repositório remoto.



Git – utilizando o log

`git log` – Verifica o histórico em ordem decrescente

`git log -p` – Mostra para cada arquivo comitado a diferença entre um arquivo e outro.

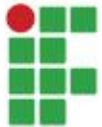
`git log -p -2` – Mostra os arquivos dos dois últimos e suas modificações.

`git log --stat` – Mostra as estatísticas de mudanças nos arquivos dos commits.

`git log --pretty=oneline` – Simplifica a visualização dos commits.

`git log --pretty=format:"%h - %an, %ar : %s"` – Mostra hash, quem comitou, quanto tempo atrás, e a mensagem.

`git log --since=2.days` – Traz o log dos dois últimos dias.



REFERÊNCIAS

- Git (git-scm.com/)
- GitHub (github.com)
- Guia prático (rogerdudler.github.io/git-guide/index.pt_BR.html)
- Controlando versões com Git e GitHub
(www.casadocodigo.com.br/products/livro-git-github)

