



Programador Web



Iniciando com controle de versão (git e github)

Adriel Sales

GIT

É um sistema de controle de versão de arquivos.

Através deles podemos desenvolver projetos no qual diversas pessoas podem contribuir simultaneamente no mesmo, editando e criando novos arquivos e permitindo que os mesmos possam existir sem o risco de suas alterações serem sobrescritas.

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



GITHUB

É uma plataforma de hospedagem de código-fonte com controle de versão usando o Git.

Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo.

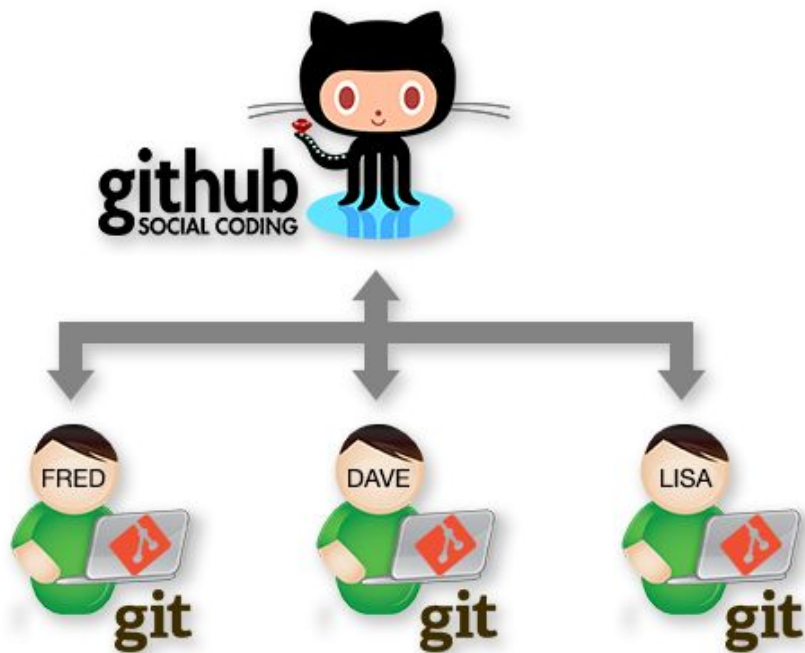
<https://github.com/>



Controle de Versão é diferente de Repositório



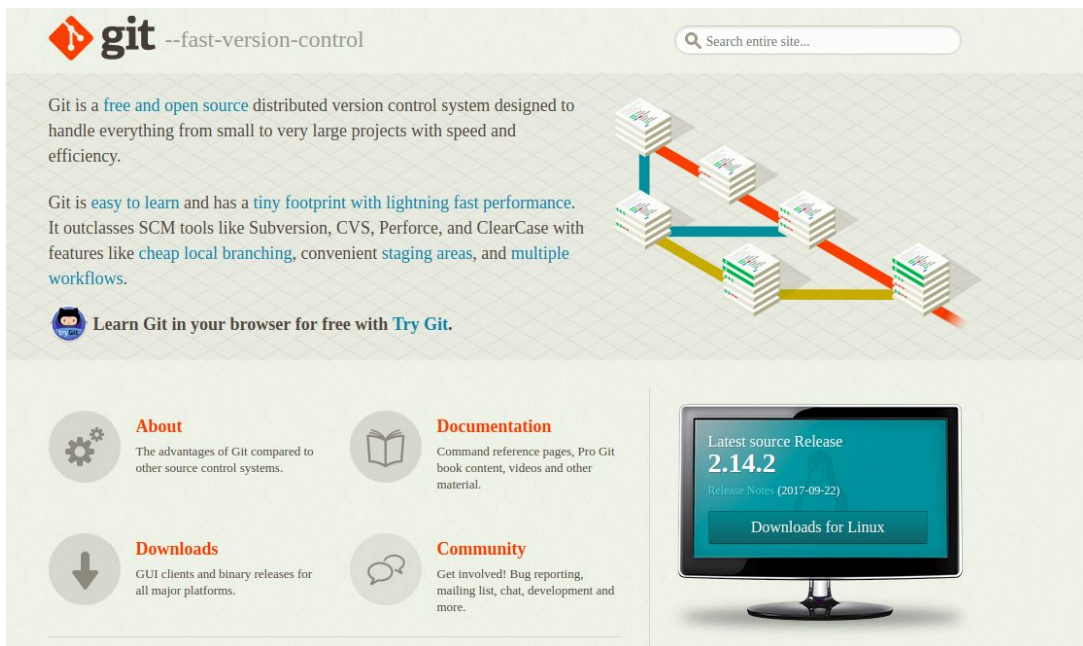
COMO FUNCIONA?



Hoje o GitHub abriga mais de **65 milhões de projetos**, alguns deles que são conhecidos mundialmente. **WordPress, GNU/Linux e o Atom.**

Passo 1:

Baixar o git e instalar em sua máquina: <https://git-scm.com/>

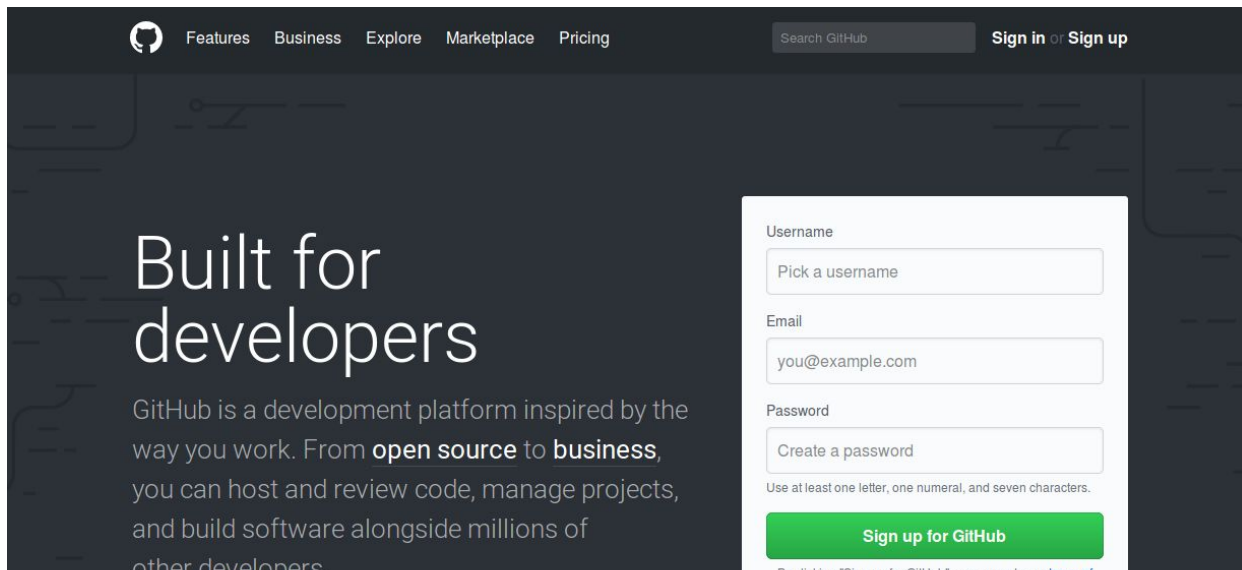


The screenshot shows the Git website homepage. At the top left is the Git logo with the tagline "--fast-version-control". To the right is a search bar. The main content area has two paragraphs: "Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency." and "Git is easy to learn and has a tiny footprint with lightning fast performance. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like cheap local branching, convenient staging areas, and multiple workflows." Below this is a GitHub logo and the text "Learn Git in your browser for free with Try Git." To the right of the text is a diagram showing a branching model with stacks of code blocks connected by colored lines (blue, red, yellow). At the bottom, there are four sections: "About" (The advantages of Git compared to other source control systems.), "Documentation" (Command reference pages, Pro Git book content, videos and other material.), "Downloads" (GUI clients and binary releases for all major platforms.), and "Community" (Get involved! Bug reporting, mailing list, chat, development and more.). On the right side of the bottom section is a monitor displaying "Latest source Release 2.14.2", "Release Notes (2017-09-22)", and a button "Downloads for Linux".

Após o download,
Next, Next, Next...

Passo 2:

Criar sua conta no github: <https://github.com/>



The screenshot shows the GitHub homepage with a dark theme. On the right side, there is a white sign-up form. The form includes fields for Username, Email, and Password, each with a placeholder text. Below the Password field, there is a note about password requirements. At the bottom of the form is a green button labeled 'Sign up for GitHub'.

Username
Pick a username

Email
you@example.com

Password
Create a password

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

Sign up for GitHub

Lembrar de aceitar o cadastro do github que foi enviado para o seu email.

Passo 3: Configuração inicial do git

1. É necessário adicionar nome e email.
2. **(use os mesmo do github):**


```
$ git config --global user.name "John Doe"  
$ git config --global user.email johndoe@example.com
```

3. Verificando as configurações:

```
$ git config --list  
user.name=Scott Chacon  
user.email=schacon@gmail.com  
color.status=auto  
color.branch=auto
```


Criando um repositório no github

1. Acesse sua conta no github.



Sign in to GitHub

Username or email address

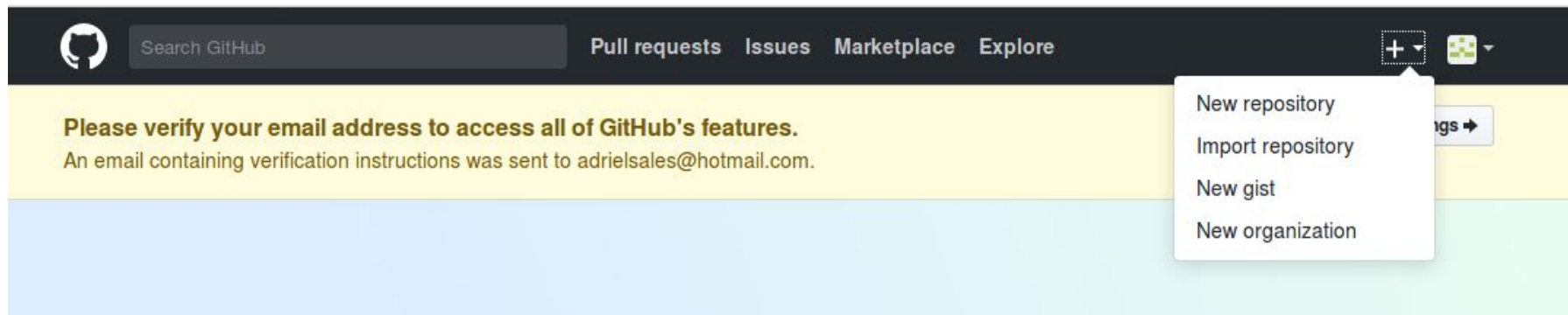
Password [Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

Criando um repositório no github

1. Clique em + e new repository.



Criando um repositório no github

Informe:

1. Nome do repositório;
2. A descrição;
3. Tipo (publico - free \$);
4. E clique em **criar!**

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner



adrielacademico ▾

Repository name

loja-programador-web



Great repository names are short and memorable. Need inspiration? How about **glowing-palm-tree**.

Description (optional)

Projeto de uma Loja Virtual para o Curso Programador Web do Senac PB (Unidade João Pessoa) 2017.



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **None** ▾



Create repository

Clonando um repositório do github para o seu PC

Projeto de uma Loja Virtual para o Curso Programador Web do Senac PB (Unidade João Pessoa) 2017.

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

adrielacademico Initial commit

README.md

Initial commit

README.md

Clone with HTTPS

Use SSH

Use Git or checkout with SVN using the web URL.

`https://github.com/adrielacademico/loja`



Download ZIP

loja-programador-web

Projeto de uma Loja Virtual para o Curso Programador Web do Senac PB (Unidade João Pessoa) 2017.

Clonando um repositório do github para o seu PC

1. Navegue até a pasta www do wamp ou htcds no lampp:

```
adrielsales@informatica01:~$ cd /opt/lampp/htdocs/
```

2. Crie um diretório para guardar os seus projetos:

```
adrielsales@informatica01:/opt/lampp/htdocs$ mkdir projetos-senac-2017
```

3. Acesse o diretório criado:

```
adrielsales@informatica01:/opt/lampp/htdocs$ cd projetos-senac-2017/
```

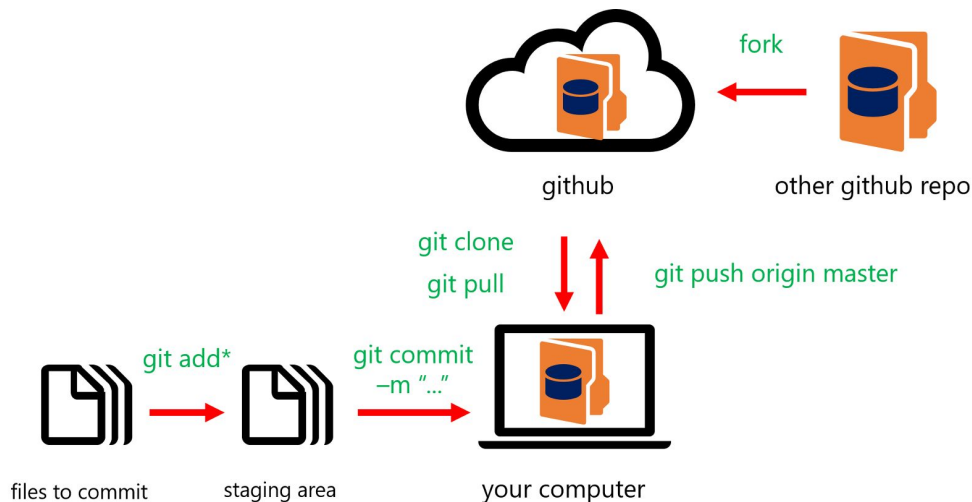
4. Clone o repositório remoto para a sua máquina:

```
adrielsales@informatica01:/opt/lampp/htdocs/projetos-senac-2017$ git clone  
https://github.com/adrielacademico/loja-programador-web.git  
Cloning into 'loja-programador-web' ...  
remote: Counting objects: 3, done.
```

Comandos mais usados no git

- Clonando um repositório do github:

```
git clone https://github.com/adrielacademico/repo-name.git
```



Comandos mais usados no git

- Iniciar um repositório git: `git init`
- Baixar atualizações no repositório do github: `git pull origin master`
- Verificar o status do seu repositório: `git status`
- Adicionar arquivos ao controle de versão: `git add .`
- “Comitar” os arquivos: `git commit -m “mensagem”`
- Enviar arquivos para o github: `git push origin master`

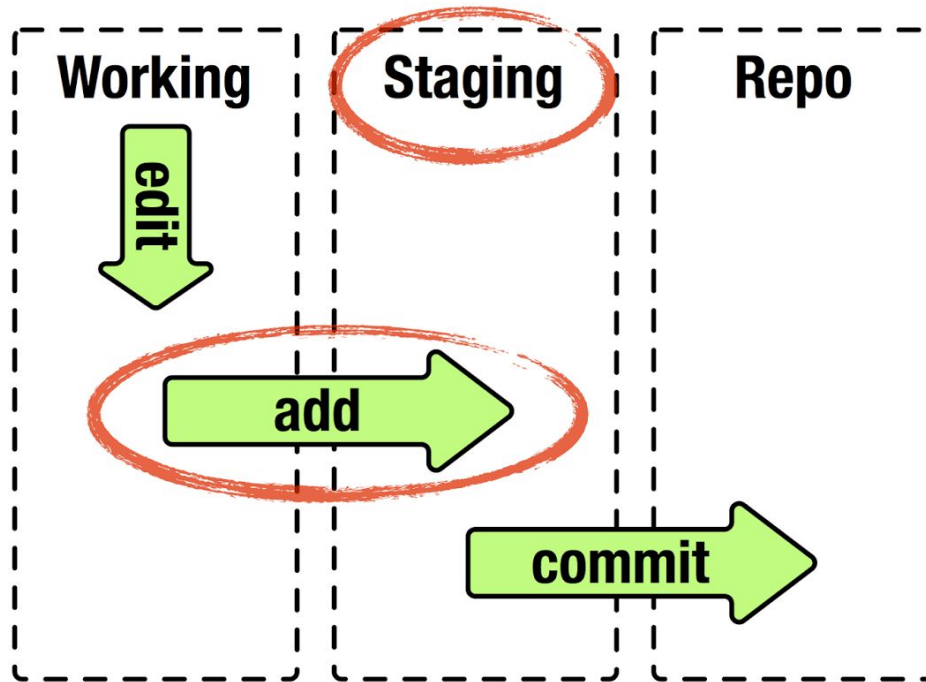
Sincronizando um repositório remoto: git → github

- Sincronizando com um repositório remoto: **git** → **github**:

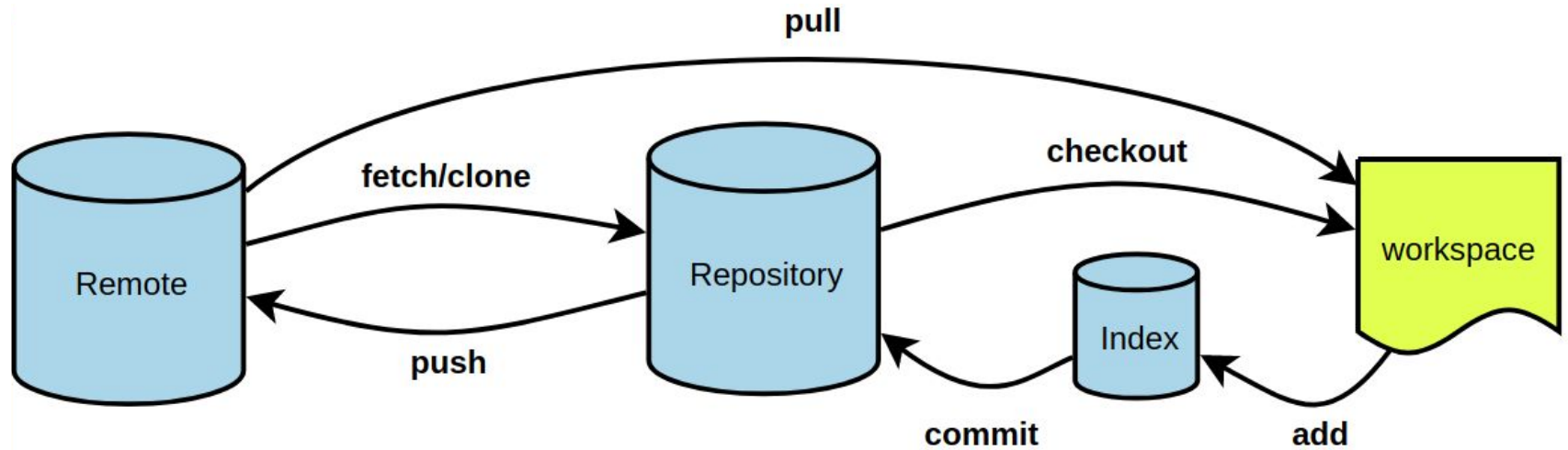
```
git remote add origin https://github.com/adrielacademico/repo-name.git
```



Estágios do git



Fluxo git e github



Cursos Free!!



<https://www.youtube.com/watch?v=WVLhm1AMeYE&list=PLInBAd9OZCzzHBJjLFZzRI6DgUmOeG3H0>



<https://www.udemy.com/git-e-github-para-iniciantes/>

Mão na massa!!

Apontamentos da Aula: <https://github.com/adrielsales/senac/wiki/Aulas-Senac>