

GIT & HUB



mentoriaT3ch  
tecnologia, tendência & transformação.

## Sobre o autor

# Giovanni de Carvalho

Minha carreira iniciou em 2016 quando comecei a cursar um curso de manutenção de computadores pelo Instituto do Banco do Brasil, ali já tinha decidido que ia seguir na área da tecnologia. Entre idas e vindas em 2017 comecei a trabalhar em uma empresa de Telecom no setor administrativo e consequentemente iniciei uma graduação de Administração de Empresas na Universidade Nove de Julho. Após 2 anos de curso, estava atuando em uma multinacional e sendo destaque em sala de aula com palestras e projetos que foram patrocinados por grandes empresas como B3 e Coca-Cola.

Bom depois de tudo isso você deve estar pensando, que mais um jovem pode querer? Bom eu ainda quero muito mais e um destes enormes desejos era retornar a área de tecnologia.

Conquistei uma bolsa de estudos no centro de inovação do Facebook, para estudar programação Web, não pensei duas vezes! Pedi demissão do meu emprego para agarrar um dos meus sonhos. De lá para cá fui aluno do Senai, realizei diversos cursos e estou no último ano de uma graduação de Análise e Desenvolvimento de Sistemas. Trabalhei em um das Startups mais reconhecidas do país e hoje trabalho no maior Banco da América Latina.

Tenho experiência em engenharia de software com integração e modelagem de sistemas, transformação digital e modernização de plataformas legadas atuando com tecnologias como: Html5, Css3, Javascript, Saas, Bootstrap, Materialize, Angular, React, Spring Boot, Django, NodeJs, MySQL, Db2, Oracle, Mainframe, Mvc, Rest, Docker. Além de experiência em: administração de equipes, administração de processos gerando KPI's relatórios e análise de dados para melhorias contínuas.



“Feito é melhor do que perfeito” e é por isso que uma ideia pior do que a sua está fazendo sucesso



- Mark Zuckerberg



# Índice Geral

visão geral, introdução

diferença de git & github

comandos cmd

instalação do git

desafio start git & github



# START

MÓDULO 1

visão geral, introdução

---

## GIT & GITHUB

# Bóra entender um pouco sobre versionamento de código

Ao final deste módulo, você será capaz de compreender os sistemas de controle de versões e compreender como o Git funciona.

Você já reparou que tudo que passa por algum tipo de atualização (ou melhoria), com o tempo possui versões diferentes? Isso ocorre com todo tipo de projeto, de produtos, carros, aplicativos, documentos etc. Em alguns casos, temos diferentes versões em funcionamento simultâneo, como, por exemplo, os consoles de vídeo game (Playstation 1, 2, 3 e 4), celulares (Iphone 6, 7, 8, X, XR), carros (Cobalt XL, XLS e XLX) e Aplicativos (Word 5, 6, 2013, 365). Geralmente, cada nova versão traz melhorias, acréscimos ou exclusão de funcionalidades.

Como eu vou utilizar o versionamento no meu dia a dia como programador(a)?

Vamos começar imaginando um cenário em que você está criando um site, independente da linguagem e demais tecnologias envolvidas na construção dele. O site vai para o ar\*. Um dia o seu cliente ou o seu chefe te recomenda (ou solicita) uma grande modificação no site, excluindo muitos dos conteúdos e adicionando coisas novas. Você executa as alterações em vários arquivos e alguns dias depois coloca essa nova versão no ar, excluindo a versão antiga.

## Sistema de controle de versões

Vamos entender como um sistema de controle de versões funciona. Existem diferentes tipos de serviços de versionamento, mas o GIT é o mais abertamente utilizado, pois é totalmente gratuito e open source\*, desenvolvido pelo mesmo criador do Sistema Operacional Linux, o Linus Torvalds.

# START

MÓDULO 2

diferença de git &  
github

---



## GIT & GITHUB

# Quais as diferenças do Git & Github

## O que é o Git?

O Git é um sistema de controle de versão de arquivos. É um software livre e muito utilizado no desenvolvimento de software onde diversas pessoas estão contribuindo simultaneamente, podendo criar e editar arquivos. Sempre quando alguém disponibiliza sua parte do projeto no Git, ele gerencia as alterações feitas e guarda um histórico. Isso é importante pois se houver algum problema você pode desfazer as alterações e voltar para a versão que estava estável.

## O que é o GitHub?

O GitHub é uma plataforma onde você pode armazenar seus projetos. É como se fosse uma rede social, só que de códigos, onde seus desenvolvedores podem disponibilizá-los para outras pessoas verem.

Quando seu projeto está no GitHub, você pode facilmente baixar uma cópia em outro computador. É uma plataforma gratuita e armazena milhões de projetos, tanto open source, pessoais e até mesmo comerciais. Alguns projetos bem conhecidos são o WordPress e o Linux. Hoje o GitHub pertence a Microsoft, que o adquiriu no ano de 2018.

Muitas pessoas utilizam o GitHub como portfólio de seu trabalho para que empresas possam vê-lo. Se você ainda não possui um GitHub e quer fazer parte, basta criar uma conta e um repositório.

# START

MÓDULO 3

comandos cmd

---



# O que é o cmd

O Prompt de Comando é um aplicativo de linha de comando disponível na maioria dos sistemas operacionais Windows, e é usado para executar os comandos inseridos. Esses comandos tem como função automatizar tarefas por meio de scripts e arquivos em lotes, executando desde funções administrativas avançadas até a solução de certos tipos de problemas do sistema operacional. O Prompt de Comando também é conhecido como shell de comando, ou prompt do cmd, ou ainda cmd.exe.

---



## Comandos cmd

`mkdir` - cria uma pasta  
`Sql server`  
`touch index.html script.js style.css` - Cria arquivos  
`cd + nome do arquivo` - entra no arquivo  
`cd ..` volta um diretório (pasta)  
`code .` para abrir o Visual Studio Code

# START

MÓDULO 4

instalação do git

---

## GIT & GITHUB

# Instalação do GIT & GITHUB

Antes de colocar a mão na massa, precisamos que você tenha o programa instalado na sua máquina.

## GIT - instalação

GIT

<https://git-scm.com/book/pt-br/v2/Come%C3%A7ando-Instalando-o-Git>

## GITHUB - cadastro

GITHUB

<https://github.com/>

O Github é uma rede social então sendo assim não é necessária a instalação. Você precisa apenas criar sua conta e utilizar



# START

MÓDULO 5

desafio start git & github

---

# Desafio

## #startGit&Github

Após todos esses conceitos vamos por a mão na massa e ir para prática.

Bom como vocês já sabem, vamos precisar abrir nosso app do git e nosso github.

## Como vai funcionar?

Dev,

Vamos subir para o github o nosso projeto já desenvolvido com Html5 e Css3.  
Nosso currículo online.

## Criando um repositório local

Para acompanhar o estado de um arquivo, por exemplo, e registrar suas versões, é necessário que o sistema de controle de versões tenha um local para armazená-lo e organizá-los. Dessa maneira, quando perguntamos qual o status atual do arquivo ou qual foi a última alteração feita nele, o sistema, neste caso o GIT, irá nos dar uma resposta com essas informações.

Esse mecanismo é chamado de repositório local. Nele serão salvas todas as versões e demais alterações que realizarmos em nosso projeto monitorado pelo GIT. Para iniciar um repositório local, você deve ir até a pasta do projeto e abrir o terminal (no caso do Windows usamos o git bash), em seguida digite o comando:

# Código no terminal Git

---

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

```
git config --global user.email = "seu email"
```

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: no email was given and auto-detection is disabled

**error** An unexpected error occurred: "Command failed.

Exit code: 1

Command: sh

Arguments: -c react-scripts build && gh-pages -d build

Directory: /Users/panda/dev/react/straydrop.github.io

Output:

"

Ele informa que foi configurado seu usuário e seu e-mail de acordo com seu cadastro na plataforma do Github.



# Código no terminal Git

---

## git init

```
+ projetogit git:(master) x git init
Initialized empty Git repository in /home/myworkspace/Área de Trabalho/projetogit/.git/
```

Ele informa que iniciou um repositório GIT vazio na pasta do meu projeto, nesse caso, "projetogit". Ao final do endereço do meu projeto, o .git, indica uma pasta nova, que nesse caso é o nosso repositório local. Para visualizá-lo graficamente, é necessário clicar com o botão direito do mouse dentro da pasta do projeto e selecionar a opção que permite exibir os arquivos ocultos.

## git status

```
@:~/week-4-game <master>$ 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)

        modified:   assets/css/style.css
        modified:   index.html

Unmerged paths:
  (use "git reset HEAD <file>..." to unstage)
  (use "git add <file>..." to mark resolution)

        both modified: assets/javascript/game.js

@:~/week-4-game <master>$
```

Exibe o status atual do repositório, tanto com arquivos editados como quanto aos arquivos já adicionados.

# Código no terminal Git

---

git add .

```
~/emacs.d(master) $> git init
Reinitialized existing Git repository in /home/nick/.emacs.d/.git/

~/emacs.d(master) $> git add -A

~/emacs.d(master) $> git status
On branch master
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   .gitignore
        modified:   auto-save-list/.save~8600~nick-ThinkPad-X200~
        modified:   projectile.cache

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)
  (commit or discard the untracked or modified content in submodules)

        modified:   el-get/ace-jump-mode (untracked content)
        modified:   el-get/ace-window (untracked content)
        modified:   el-get/dash (untracked content)
        modified:   el-get/f (untracked content)
        modified:   el-get/flycheck (untracked content)
        modified:   el-get/fuzzy (untracked content)
        modified:   el-get/s (untracked content)

~/emacs.d(master) $> █
```

Adiciona todos os arquivos que estão sendo editados, caso preferir pode adicionar apenas um arquivo por vez, como por exemplo:  
git add arquivo

# Código no terminal Git

---

`git commit -m "mensagem_commit"`

```
HiManshu@HiManshu-PC MINGW64 ~/Desktop/NewDirectory (master)
$ touch newfile4.txt

HiManshu@HiManshu-PC MINGW64 ~/Desktop/NewDirectory (master)
$ git status
on branch master
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:   newfile3.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        newfile4.txt

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

HiManshu@HiManshu-PC MINGW64 ~/Desktop/NewDirectory (master)
$ git commit -a
[master fc66f84] updated newfile3
1 file changed, 1 insertion(+)
```

Registra todos os arquivos que foram adicionados, caso preferir  
pode usar o comando, como por exemplo:  
`git commit -a`

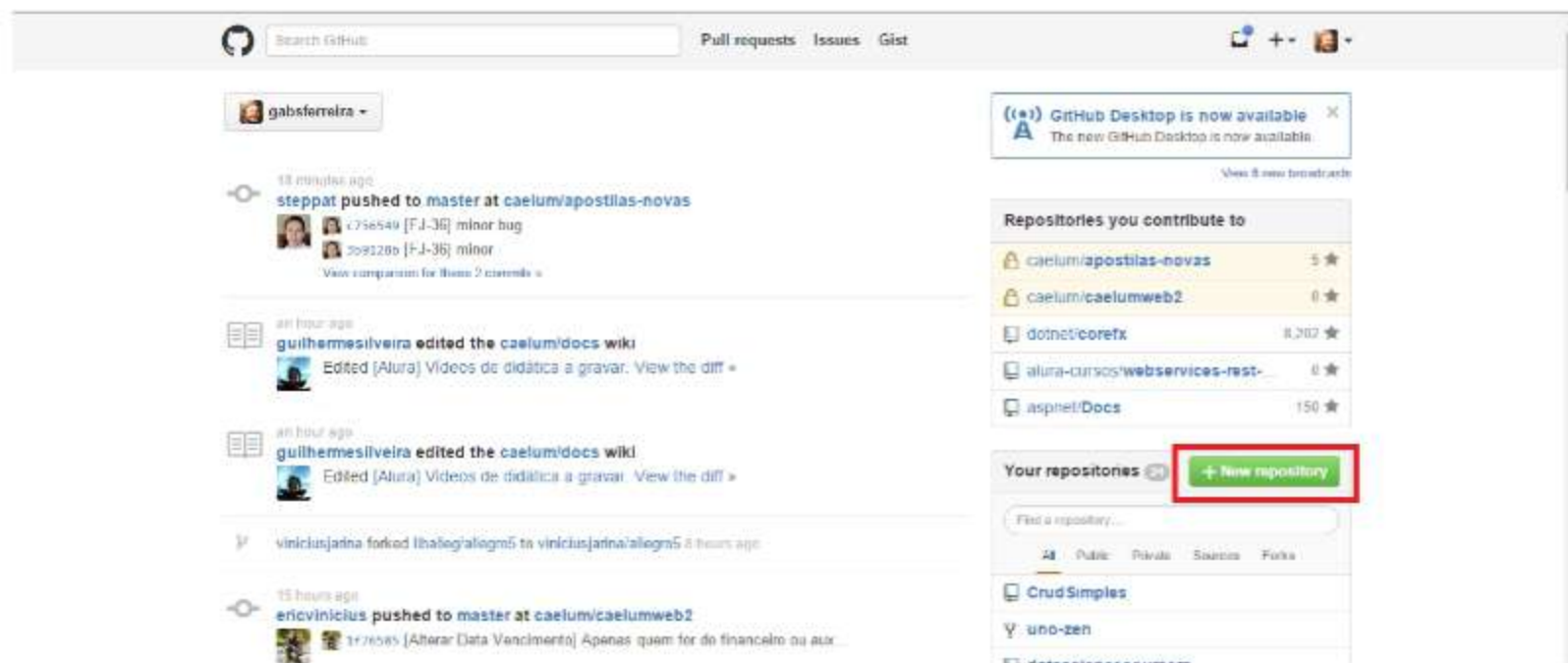


# Código no terminal Git

## Criando o repositório

Você precisará agora criar seu repositório remoto no Github, para que todo seu código já iniciado e registrado suba para aquele servidor.

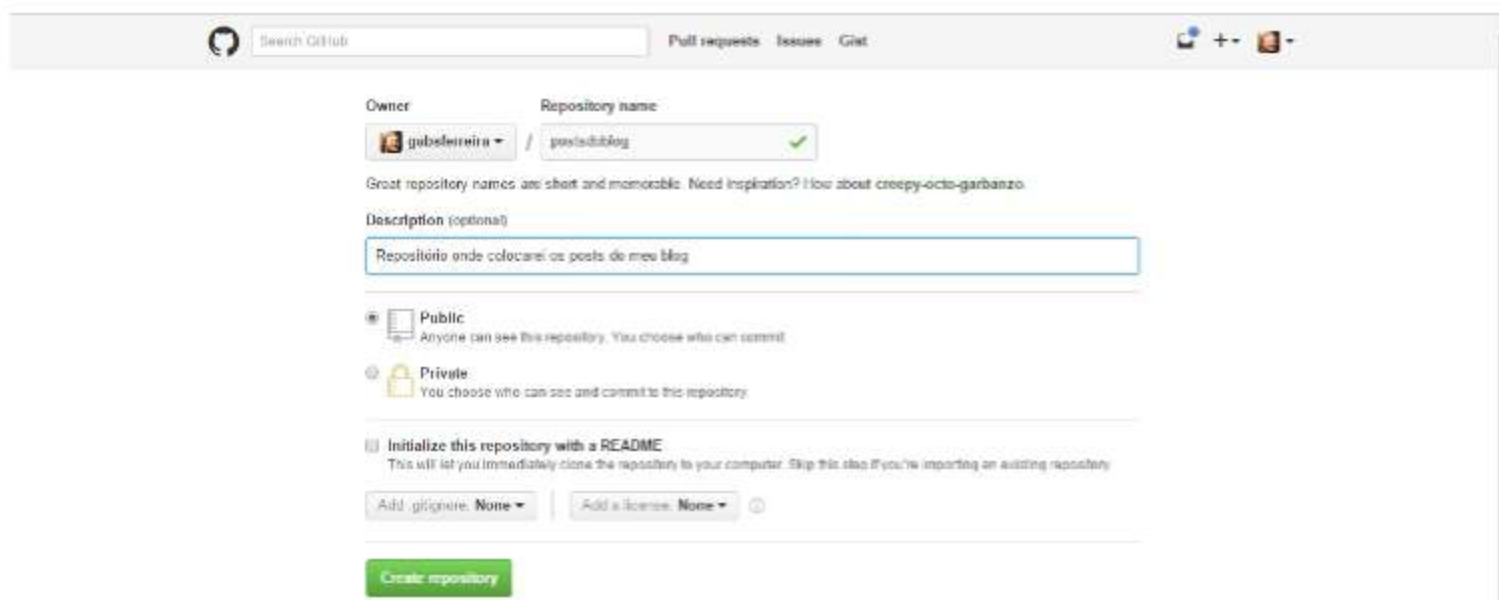
Abra o Github, faça login com a sua conta e clique em New repository.



# Código no terminal Git

---

## Criando o repositório



The screenshot shows the GitHub 'Create repository' page. At the top, there's a search bar and navigation links for 'Pull requests', 'Issues', and 'Git'. The main form has two input fields: 'Owner' (set to 'gubsterreira') and 'Repository name' (set to 'possidiblog' with a green checkmark). Below these, a hint says 'Great repository names are short and memorable. Need inspiration? How about creepy-octo-garbanzo.' The 'Description (optional)' field contains 'Repositório onde colocarei os posts do meu blog'. There are two radio button options: 'Public' (selected) and 'Private'. Below these is a checkbox for 'Initialize this repository with a README'. At the bottom, there are dropdown menus for 'Add a license' (set to 'None') and 'Add a license' (set to 'None'). A green 'Create repository' button is at the bottom.

Owner: gubsterreira / Repository name: possidiblog

Great repository names are short and memorable. Need inspiration? How about creepy-octo-garbanzo.

Description (optional): Repositório onde colocarei os posts do meu blog

☒ 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.

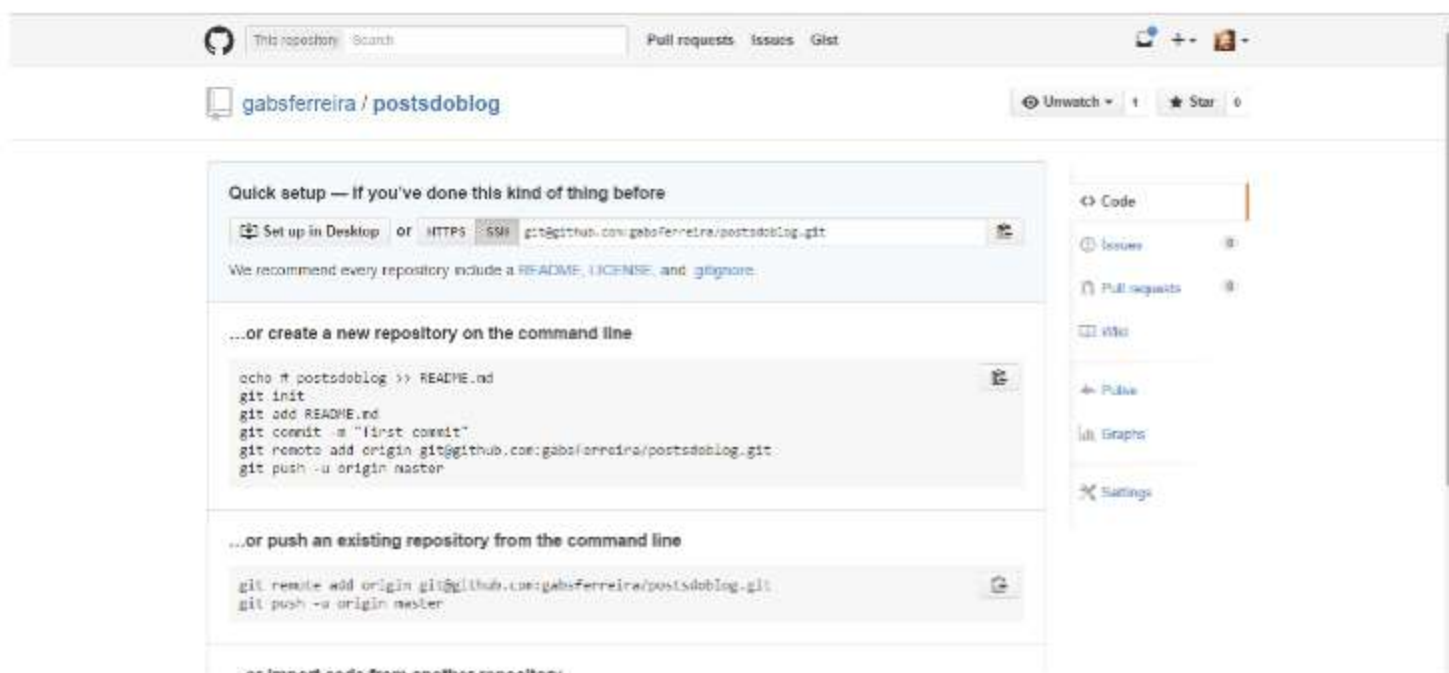
Add a license: None

Create repository

Em seguida, basta colocar o nome e descrição do repositório que você está criando e clicar em Create repository.

# Código no terminal Git

## Criando o repositório



Pronto! O repositório foi criado, só que ainda está vazio.



# Código no terminal Git

---

git remote add origin "seu link do repositório do github"

```
DELL@Buwaneka MINGW64 ~/workspace/Eclipse Neon/Medic (master)
$ git remote add origin https://github.com/BSThisarasinghe/Sabra-Medical-Center.git
fatal: remote origin already exists.
```

Registra suas adições no seu repositório local do github.

git push origin master

```
Welcome@Welcome-PC MINGW64 /e/git-demos/push-test (master)
$ git push origin master
Counting objects: 3, done.
Delta compression using up to 4 threads.
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 302 bytes | 302.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0)
To https://github.com/git-test-jaz/push-tst.git
45c8cf7..8aac6ed master -> master
```

Emfim sobe suas alterações ao seu repositório no github.



# Obrigado

Queremos agradecer todo seu empenho e disposição em participar deste projeto que é a mentoria tech uma comunidade construída de alunos para alunos com grandes ambições e objetivos dentre eles temos:

- Fazer com que os alunos alcancem altos níveis em suas formações.
- Mentoria de alunos experientes para apoiar e moldar os alunos mais novos.
- Agregar valor na busca de seu primeiro emprego.
- Produzir e compartilhar conteúdo de qualidade feito por alunos com foco em alunos.
- Disseminar e promover acessibilidade e igualdade em nossos conteúdos ajudando assim toda cota de alunos.
- Realizar encontros com outras comunidades e eventos que gerem impactos positivos como networking, experiências, preposições técnicas entre outros.

# Busque nossas redes sociais



+55 (11) 9 61985346



contatomentoriatech@gmail.com



@mentoriatech



[www.mentoriatech.com.br](http://www.mentoriatech.com.br)

## Bootcamp



ACESSE AGORA

Siga nossas mídias sociais  
[@mentoriatech](#)