

MINICURSO

GITHUB + VERCCEL

ANÁLISE E DESENV.
DE SISTEMAS



Prof^a Letícia Pieper Jandt



O que é Github

GitHub é uma plataforma de hospedagem de código-fonte e arquivos 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.



O GIT também funciona como um portfólio...
onde você pode mostrar seus projetos a
qualquer momento.

Somada a vercel, você pode fazer uma
apresentação a qualquer momento.



Por que fazer o versionamento de código?

Por que fazer backup?

PORQUE EU PRECISO SABER DISSOOOOO?



Passo a passo para usar o git

- Arquivos importantes: .gitignore

O Git vê cada arquivo em sua cópia de trabalho em um de três estados:

1. rastreado – um arquivo que já passou pelo staging ou commit;
2. não rastreado – um arquivo que *não* passou pelo staging ou commit;
3. ignorado – um arquivo que o Git foi forçado a ignorar.



Passo a passo para usar o git

- Os arquivos ignorados são rastreados em um arquivo especial chamado *.gitignore*, que é verificado na origem do seu repositório.
- É preciso editar e fazer commit do arquivo *.gitignore* à mão.
- Os nomes dos arquivos do projeto são comparados aos nomes do arquivo de configuração do *.gitignore*.



Passo a passo para usar o git

- Arquivos importantes: README.md

1. README.md é um arquivo com extensão .md (Markdown). Contém informações necessárias para entender o objetivo do projeto. README é uma palavra em *inglês* que traduzida fica LEIAME.
2. Podemos considerar o README como um cartão de visita do seu projeto no Github ou em outras plataformas de repositórios remotos de código: Gitlab, Bitbucket, Azure Repos.



Passo a passo para usar o git

Termos usados: Commit

Refere-se ao processo de tornar permanente um conjunto de alterações, ou seja, de efetivar as alterações. Um uso comum é a conclusão de uma transação.

Sempre que faço uma alteração, preciso fazer um commit com os arquivos alterados!



Passo a passo para usar o git

Termos usados: Master

O termo “master” é utilizado para apontar o principal ramo de um repositório.

Pasta Raiz do projeto.



Passo a passo para usar o git

Termos usados: Origin

O “origin” é o nome do repositório remoto padrão em que se criou localmente através do clone.

Repositório original.



Passo a passo para usar o git

Principais comandos:



Git Init

Para começar um projeto que ainda não seja um repositório no git, o comando **Git Init** costuma ser o comando mais indicado.

Basicamente, ele cria um repositório vazio ou transforma uma pasta que você já tem e que não está com controle de versão em um repositório.



Git Add

Quando criamos, modificamos ou excluimos um arquivo, essas alterações ocorrerão em nosso ambiente local e não serão incluídas no próximo *commit* (a menos que alteremos as configurações).

Para incluir as alterações de um arquivo em nosso próximo *commit*, será preciso usar o comando **git add**.

Para adicionar apenas um arquivo:

git add <arquivo>

Para adicionar, de uma vez, todos os arquivos modificados:

git add -A

Vale lembrar que o comando **git add** não altera o repositório e as alterações não são salvas até usarmos o *git commit*.



Git commit

Esse comando cria uma versão de código, para o qual você pode voltar mais tarde, se necessário.

Útil quando ocorrem erros no projeto, podendo voltar uma versão funcional enquanto é feita a correção do mesmo.

`git commit -m "mensagem explicando a mudança no código"`



Git branch

O uso de branch's serve para criar “braços”, ou seja, serve para criar blocos de códigos separados, assim, várias pessoas podem mexer nesses braços e o projeto principal (main) não será alterado.

Assim, quando as alterações das branch's acabam, é feito commit no projeto principal.



Git branch -M main

Para commitar as alterações das outras branches na master é usado outros comandos como **merge**, **pull request**, **rebase**, porém, essa parte avançada deixamos para o minicurso parte 2.





Git branch -M main

O comando abaixo cria uma ramificação chamada main, que será nossa branch principal:

git branch -M main

o repositório será a raiz, a master.



Git remote add origin

O comando **git remote add** criará um novo registro de conexão para um repositório remoto. Depois de adicionar um controle remoto, você poderá usá-lo como um atalho conveniente para outros comandos do Git.

Ou seja, cria um “túnel de conexão” com o repositório.



Git push

Após confirmar as alterações, a próxima coisa que você deseja fazer é enviar as alterações para o servidor remoto usando o comando

git push



Passo a passo para usar o git

Depois de instalado, devemos configurar o git com nosso usuário:

```
git config --global user.name "João Silva"
```

```
git config --global user.email "exemplo@seuemail.com.br"
```

*caso nao tenha uma conta, tem que criar no github



Letícia Pieper

Pieper04

Chefe de Desenvolvimento WEB, chefe de estágio desenvolvimento, professora universitária e entusiasta de tecnologia. Especialista em desenvolvimento de sistemas

[Edit profile](#)

10 followers · 4 following

RuralHub

Sinop - MT

leticia.pieper1@gmail.com

ruralhub.com.br

Popular repositories

[Customize your pins](#)

phonegap-plugin-push

Public

Forked from [phonegap/phonegap-plugin-push](#)

Register and receive push notifications

Objective-C

SiloVirtual2

Public

conads2023

Public

Minicurso ministrado no CONADS 2023

JavaScript

Conads2023_1

Public

JavaScript

DispMoveis

Public

Path das aulas disp. móveis 2023/2

JavaScript

POO2

Public

CÓDIGOS POO 2 2023/2

Java

150 contributions in the last year

[Contribution settings](#)

2024



2023

2022

2021

2020



Acessar o Git e criar um novo repositório



Pieper04



Overview



Repositories

14



Projects



Packages



Stars



Popular r

Acessar Repositories



Find a repository...

Type ▾

Language ▾

Sort ▾

 New

Clicar em new



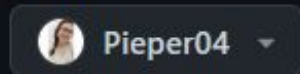
Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?

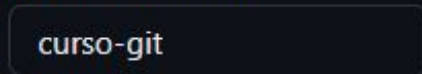
[Import a repository.](#)

Required fields are marked with an asterisk ().*

Owner *



Repository name *



✓ curso-git is available.



Preencha o nome do
projeto
que é o mesmo nome do
projeto já existente

Great repository names are short and memorable. Need inspiration? How about **super-octo-telegram** ?

Description (optional)



Public

Anyone on the internet can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.



Selecione o modo de
visualização do projeto.



Initialize this repository with:

☒ Add a README file

This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template: **None** ▼

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license

License: **None** ▼

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.

Create repository

Aqui podemos selecionar se queremos que crie automático o arquivo sobre.

Se queremos gerar um arquivo de licença



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

 Set up in Desktop or [HTTPS](#) [SSH](#) `https://github.com/Pieper04/curso-git.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 "# curso-git" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Pieper04/curso-git.git
git push -u origin main
```

Aqui o próprio git já nos dá exemplos de uso.

Vamos usar as linhas de comando.

...or push an existing repository from the command line

```
git remote add origin https://github.com/Pieper04/curso-git.git
git branch -M main
git push -u origin main
```

...or import code from another repository

You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

[Import code](#)



Criando nosso primeiro commit



Via cmd entre na pasta do projeto:

```
cd c:/....
```

1. `echo "# curso-git - meu 1º commit" >> README.md`
2. `git init`
3. `git add .` (para adicionar todos os arquivos no commit)
4. `git commit -m "Meu primeiro commit"`
5. `git branch -M main`
6. `git remote add origin 'Link do seu repositório'`
7. `git push -u origin main`





Pieper04 Meu primeiro commit

5dbe0ee · 8 minutes ago 2 Commits

public	Meu primeiro commit	8 minutes ago
src	Meu primeiro commit	8 minutes ago
.gitignore	Initialize project using Create React App	24 minutes ago
README.md	Meu primeiro commit	8 minutes ago
package-lock.json	Initialize project using Create React App	24 minutes ago
package.json	Initialize project using Create React App	24 minutes ago

README

Projeto teste - Curso GitHub.

Seja bem vindo! Caso deseje alterar alguma informação acesse:

`src/App.js`

Código gerado para curso de Git 21/02/2024 Letícia Pieper Jandt

Se tudo deu
certo, esse é o
resultado!



Upando na vercel

Vercel Inc. é uma plataforma de nuvem americana como empresa de serviços.

A arquitetura da Vercel é construída em torno do Jamstack e as implantações são feitas por meio de repositórios Git.

A Vercel é membro da MACH Alliance.



Upando na vercel

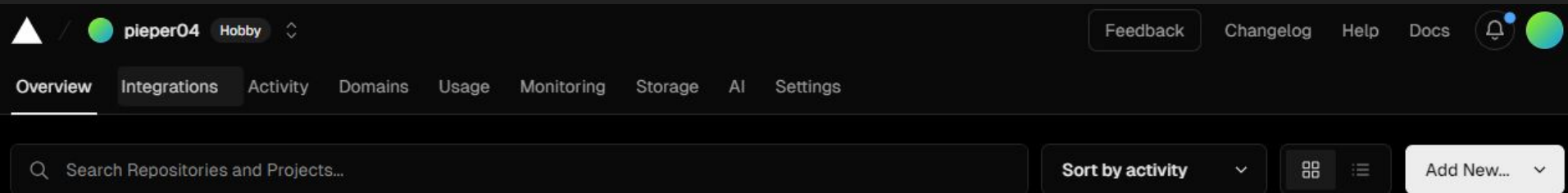
<https://vercel.com>

Crie uma conta vinculada ao GitHub




Upando na vercel


Click em Add new -> projects



Import Git Repository

 Pleper04



 Search...



curso-git · 6m ago

Import



DispMoveis · 173d ago

Import



POO2 · 170d ago


Import



ESTRUTURA-DE-DADOS · 106d ago

Import



Teste-vocacional  · 127d ago

Import





Configure Project

Project Name

conads2023

Framework Preset



Create React App



Root Directory

./

Edit

> Build and Output Settings

> Environment Variables

Deploy

Congratulations!

You just deployed a new Project to Vercel.

[Continue to Dashboard](#)

Olá! Meu nome é Leticia e eu sou professora de ADS.

Esse é meu primeiro comit GitHub.

[Análise e Desenvolvimento de sistemas - Univasipe.](#)

Next Steps



Instant Previews

Push a new branch to preview changes instantly



Add Domain



Add a custom domain to your project



Enable Speed Insights



Track how users experience your site over time



<https://curso-git-nu.vercel.app>

Meu lindo!



Alterando o código:

1. Faça uma alteração em algum arquivo.
2. Execute o comando: *git add -A*
3. Execute o comando: *git commit -m "alterando arquivos - criando versões"*
4. Execute o comando *git push*



Pieper04 alterando arquivos - criando versões



94172d1 · now



3 Commits



public

Meu primeiro commit

15 minutes ago



src

alterando arquivos - criando versões

now



.gitignore

Initialize project using Create React App

32 minutes ago



README.md

Meu primeiro commit

15 minutes ago



package-lock.json

Initialize project using Create React App

32 minutes ago



package.json

Initialize project using Create React App

32 minutes ago



A mensagem e o arquivo serão alterados.



Olá! Meu nome é Letícia e eu sou professora de ADS.

Esse é meu primeiro comit GitHub.

[Análise e Desenvolvimento de sistemas - Unifasipe.](#)

O Site da vercel também será alterado
automaticamente.

OBRIGADA E GRATILUZ!

- DÚVIDAS
- COMENTÁRIOS
- CRÍTICAS APENAS
DESTRUTIVAS (SE CONSEGUIR)

@leticiapieperjandt
github.com/Pieper04/
Linkedinho: Letícia Pieper



Conselho do dia:





Arquivos de estudo:

<https://www.hostinger.com.br/tutoriais/tutorial-do-git-basics-introducao#:~:text=Se%20você%20tem%20uma%20conta,e%20clicar%20no%20botão%20criar.>

<https://www.atlassian.com/br/git/tutorials/saving-changes/gitignore>

<https://blog.rocketseat.com.br/o-que-e-readme-e-porque-e-tao-importante/>

<https://blog.geekhunter.com.br/comandos-git-mais-utilizados/>

https://blog.geekhunter.com.br/comandos-git-mais-utilizados/#Os_commits