

INSTITUTO MAUÁ DE TECNOLOGIA



Linguagens I

GIT

Profº. Tiago Sanches da Silva

O que é GIT?

O que é GIT?

Git é um sistema de controle de versão de arquivos. Através deles podemos desenvolver projetos na qual diversas pessoas podem contribuir simultaneamente, editando e criando novos arquivos. Permitindo que os mesmos possam existir sem o risco de suas alterações serem sobrescritas.

O que é github?

O GitHub estende a ferramenta Git com uma ampla gama de recursos colaborativos e oferece hospedagem gratuita para projetos open source.

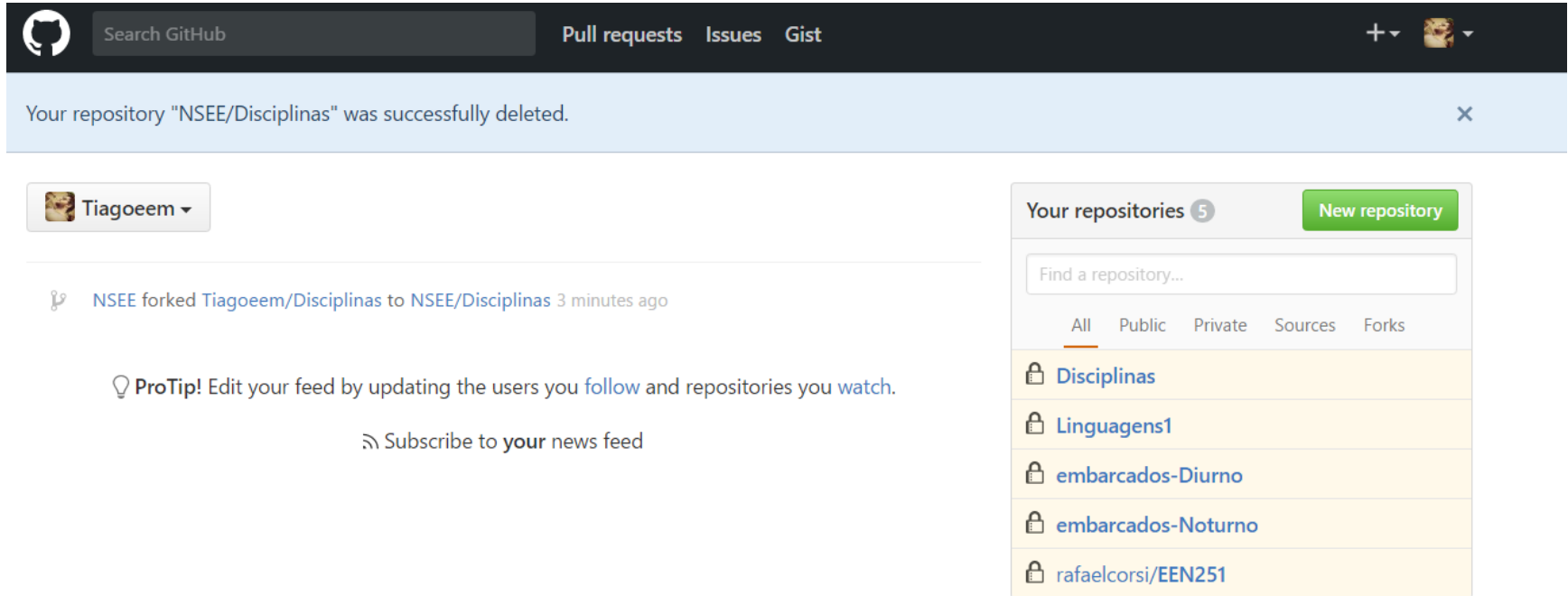


Existem outros?

SIM



Podemos explorar o Github?



The screenshot shows the GitHub web interface. At the top, there's a dark navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, and Gist. A notification banner states: "Your repository 'NSEE/Disiplinas' was successfully deleted." Below this, the user profile for "Tiagoem" is visible. A feed item shows "NSEE forked Tiagoem/Disiplinas to NSEE/Disiplinas 3 minutes ago". A "ProTip!" suggests editing the feed by following users and watching repositories. A "Subscribe to your news feed" link is also present. On the right, the "Your repositories" section shows 5 repositories: Disciplinas, Linguagens1, embarcados-Diurno, embarcados-Noturno, and rafaelcorsi/EEN251.

Search GitHub

Pull requests Issues Gist

Your repository "NSEE/Disiplinas" was successfully deleted.

Tiagoem

NSEE forked Tiagoem/Disiplinas to NSEE/Disiplinas 3 minutes ago

ProTip! Edit your feed by updating the users you follow and repositories you watch.

Subscribe to your news feed

Your repositories 5

New repository

Find a repository...

All Public Private Sources Forks

- Disciplinas
- Linguagens1
- embarcados-Diurno
- embarcados-Noturno
- rafaelcorsi/EEN251



“Sim, podemos!”

Primeira configuração

Primeira configuração

Você deverá configurar o nome e e-mail que assinarão seus *commits*.

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

Esta configuração deve ser feita apenas uma vez em cada máquina de trabalho.

Cenário 1

Cenário 1

- Criar o repositório no github.
- “Anotar” o nome do repositório **remoto**.
- Clonar na maquina local o repositório remoto. (inicializa e configura o servidor remoto)
 - ❖ Será criado uma pasta com o nome do repositório.



- Trabalhe
- Adicione o(s) arquivo para entrar no seu próximo *commit*
- Realize um *commit*
- Atualize o repositório remoto

Cenário 1

- Criar o repositório no github.

Create a new repository

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

Owner: Tiagoem / Repository name: OlaMundo ✓

Great repository names are short and memorable. Need inspiration? How about `solid-octo-guacamole`.

Description (optional)

Meu primeiro repositóriozinho :D

☒ **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

https ou ssh?



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

Set up in Desktop or **HTTPS** SSH

We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# OlaMundo" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/Tiagoem/OlaMundo.git
git push -u origin master
```

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

```
git remote add origin https://github.com/Tiagoem/OlaMundo.git
git push -u origin master
```

...or import code from another repository

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

- “Anotar” o nome do repositório **remoto**.

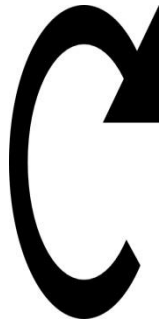
Cenário 1

- Clonar na maquina local o repositório remoto. (inicializa e configura o servidor remoto)
 - ❖ Será criado uma pasta com o nome do repositório.

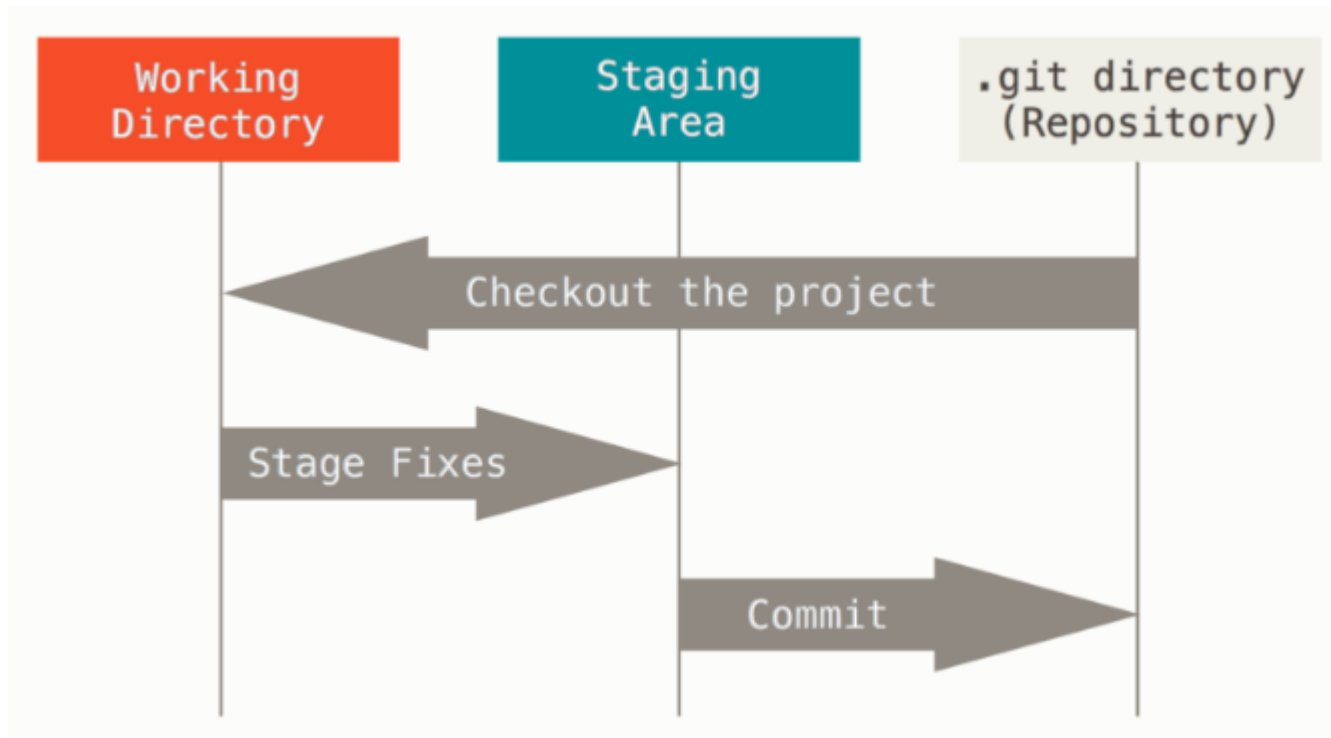
```
git clone usuário@servidor:/caminho/para/o/repositório
```

Ex: `git clone https://github.com/Tiagoeem/teste.git`

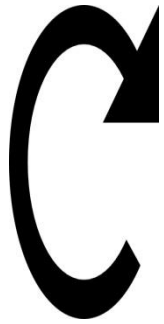
Cenário 1



- Trabalhe
- Adicione o(s) arquivo para entrar no seu próximo *commit*
- Realize um *commit*



Cenário 1



- Trabalhe
- Adicione o(s) arquivo para entrar no seu próximo commit
- Realize um *commit*

```
git add <arquivo>
```

```
git add *
```

```
git commit -m "comentários das alterações"
```

```
git status
```

```
git status
```

```
git status
```

```
git status
```

Cenário 1

git status

```
TiagoLow@TiagoLow-PC MINGW64 /e/Mauá/Disciplinas (master)
$ git status
On branch master
Your branch is up-to-date with 'origin/master'.
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:   ecm251-Linguagens/aulas/aula1/GIT.pptx

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

        ecm251-Linguagens/aulas/aula1/~$GIT.pptx

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

```
TiagoLow@TiagoLow-PC MINGW64 /e/Mauá/Disciplinas (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:   ecm251-Linguagens/aulas/aula1/GIT.pptx

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

        ecm251-Linguagens/aulas/aula1/~$GIT.pptx
```

Cenário 1

- Atualize o repositório remoto

```
git push origin master
```

Alias (“apelido”) padrão do repositório remoto

Nome do *branch* que você quer atualizar

→ <https://github.com/Tiagoeem/teste.git>

git remote

git remote

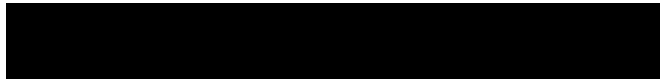
Verificar se o servidor remoto está configurado no seu repositório GIT.

```
git remote -v
```

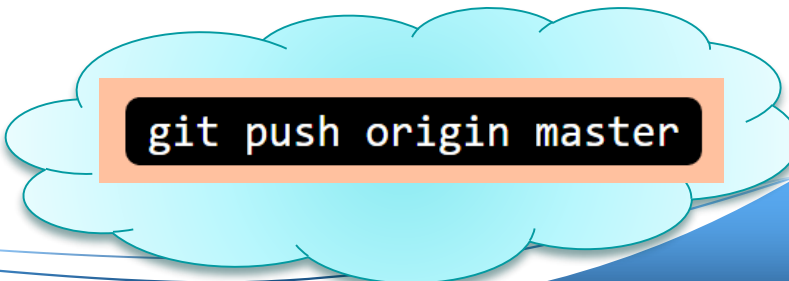
Caso esteja:

```
origin  git@github.com:Tiagoem/Disciplinas.git (fetch)  
origin  git@github.com:Tiagoem/Disciplinas.git (push)
```

Caso não:



Cada repositório **git local**, possui seu próprio **remote (server remoto)**



```
git push origin master
```

git remote

Adicionando um servidor ao seu repositório local.

```
git remote add origin <servidor>
```

Cada repositório **git local**, possui seu próprio **remote (server remoto)**

Poderia ser outro alias, mas não esqueça de usar o mesmo no **git push**

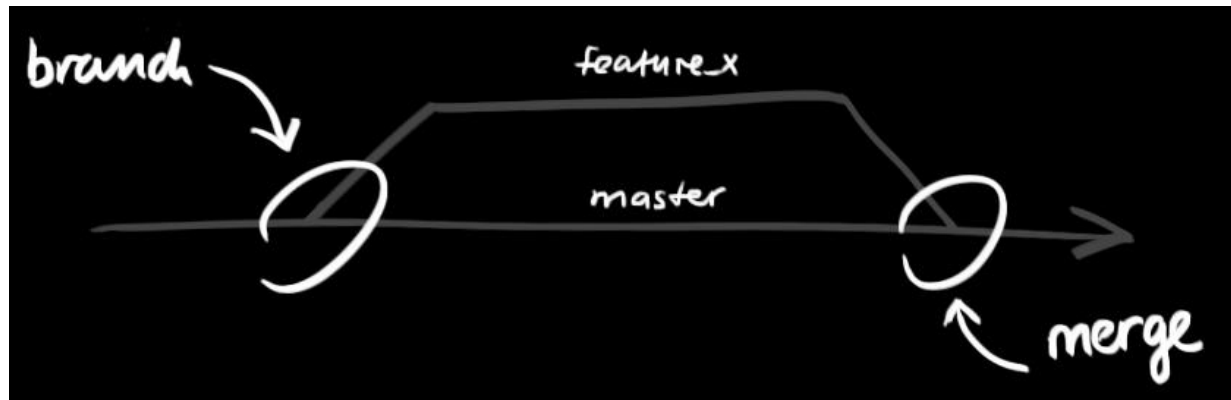
```
git push origin master
```

Ramificação

branches

Branches ("ramos") são utilizados para desenvolver funcionalidades isoladas umas das outras.

O branch master é o branch "padrão" quando você cria um repositório. Use outros branches para desenvolver e mescle-os (merge) ao branch master após a conclusão.



branches

crie um novo branch chamado "funcionalidade_x" e selecione-o usando

```
git checkout -b funcionalidade_x
```

retorne para o master usando

```
git checkout master
```

e remova o branch da seguinte forma

```
git branch -d funcionalidade_x
```

um branch *não está disponível a outros* a menos que você envie o

branch para seu repositório remoto

```
git push origin <funcionalidade_x>
```

Mergear branches distintos

Para mergear outro *branch* no seu atual ativo:

```
git merge <branch>
```

E se der conflito?

Atualizar repositório local

Atualizar repositório

Atualizar seu repositório local com a mais nova versão, disponível no repositório remoto.

```
git fetch  
git merge
```

ou

```
git pull
```

Outras ações

Liguei meu PC, vou começar a trabalhar no código.

Será que alguém subiu alguma alteração? Será que preciso dar uma atualizada no meu repositório local (git pull ou git fetch && git merge)?

O que devo fazer?



Outras ações

Liguei meu PC, vou começar a trabalhar no código.

Será que alguém subiu alguma alteração? Será que preciso dar uma atualizada no meu repositório local (git pull ou git fetch && git merge)?

Verifique!

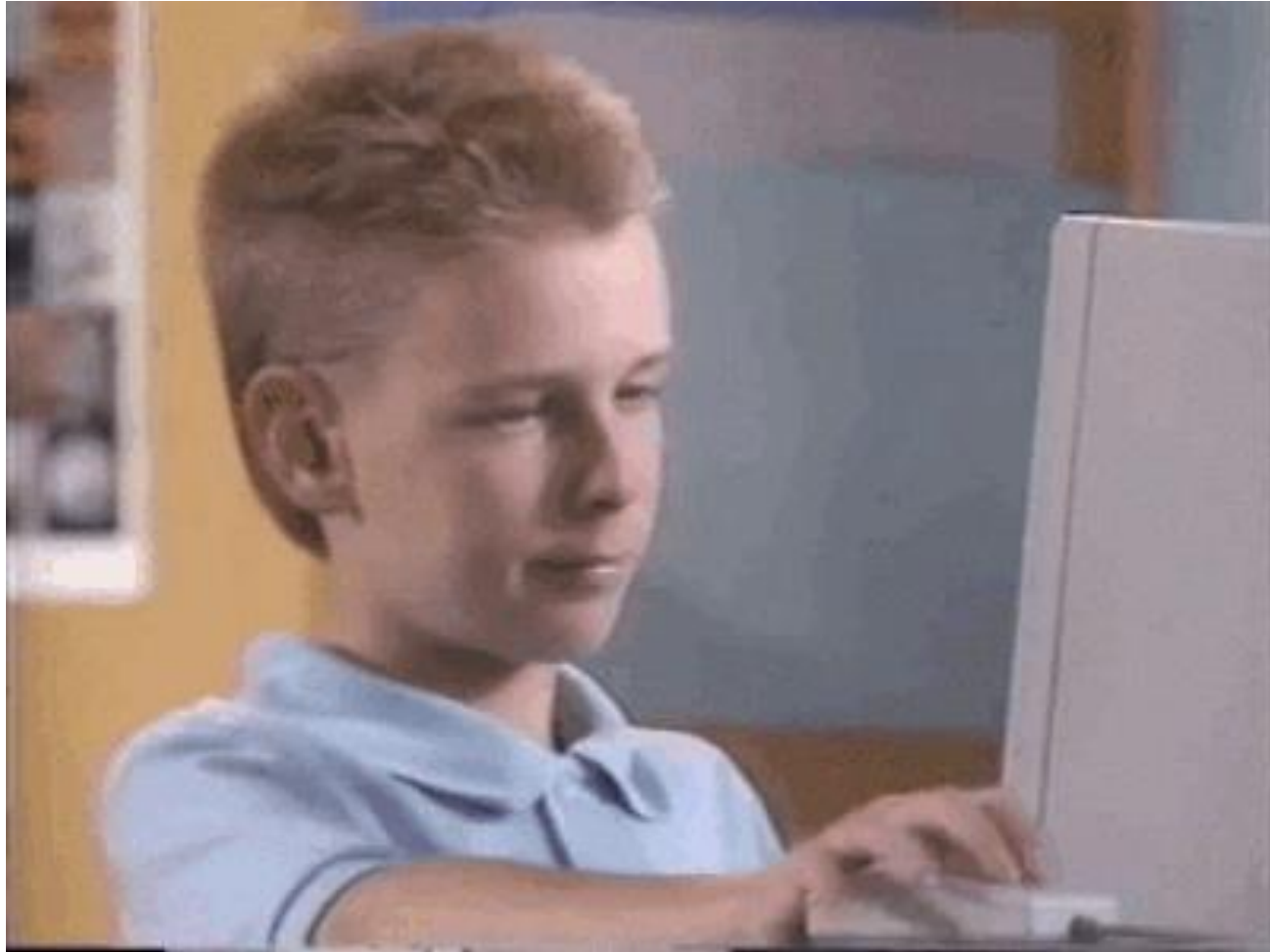
```
git remote update && git status
```

Cenário 2

Cenário 2

- Iniciar repositório git local: `git init`
- E depois?

Vamos praticar?



Exercício

- Para que serve o git log?
- E o git reset?
- git ignore?
- Como remover o ultimo commit?
- Como voltar a um commit especifico?
- Qual a diferença entre git pull e git push?
- Quando devo deletar a pasta .git do meu repositório local?

Perguntas?