

Git e GitHub

Prof. Jesiel Viana
@jesielviana

Objetivos de aprendizagem

- Conhecer o Git
- Entender a necessidade e vantagens do Git
- Conhecer o GitHub
- Aprender os comandos básicos do Git
- Criar um projeto no GitHub
-

Contexto

- O desenvolvimento de software geralmente é realizado em equipe, com a colaboração de vários desenvolvedores, que em muitos casos estão trabalhando em um mesmo projetos de locais e horários diferentes uns dos outros;
- Software é desenvolvido seguindo o sistema de versões, 1.0, 1.1, 2.0, etc;
- Software precisa de manutenção.

Problema

- Como juntar o código de diversos programadores em um mesmo projeto?
- E se mais de um programador alterar um mesmo arquivo?
- Como separar o código da versão que está em produção da versão que está em desenvolvimento?
- Como manter um histórico de modificações do software, do arquivo?
- Como Gerenciar as versões?

Git

Sistema de controle de versão que permite integrar as modificações de diversos colaboradores em um única base de código, fazendo o controle de modificações e conflitos. Além disso, cria todo histórico de alterações no código do projeto, permitindo facilmente voltar para qualquer ponto para saber como o código estava naquele momento [1].

Git

É um sistema open-source de controle de versão, sendo o mais utilizado pelos desenvolvedores atualmente. Foi desenvolvido em 2005 por Linus Torvalds, o famoso criador do kernel do sistema operacional Linux [2].

[Repositório do Git](#)

Github

GitHub é uma plataforma de hospedagem de código (repositórios Git) para controle de versão e colaboração. Ele permite que você e outras pessoas trabalhem juntos em projetos de qualquer lugar [4].

Além de servir como hospedagem, o Github possui muitas integrações com serviços que auxiliam no deploy da aplicação através de integração contínua [3].

Principais Comandos Git

init e clone

- o comando **git init** cria um repositório vazio.

```
git init
```

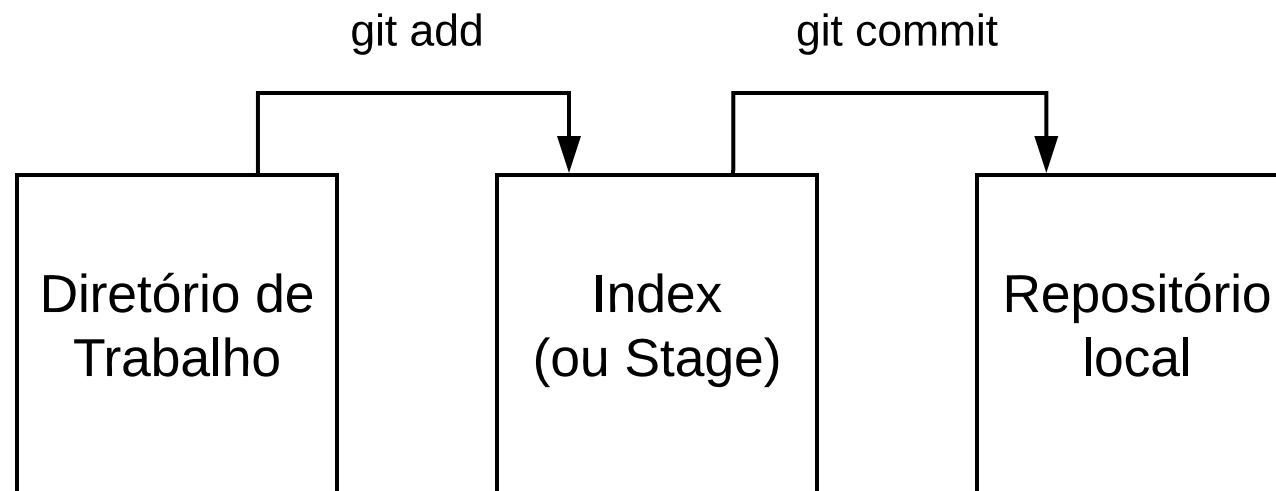
- comando **git clone** clona(copia código e histórico) para o diretório corrente(local) um repositório armazenado remotamente, por exemplo: um repositório do GitHub.

```
git clone https://github.com/NOME-USER/NOME-REPO
```

add e commit

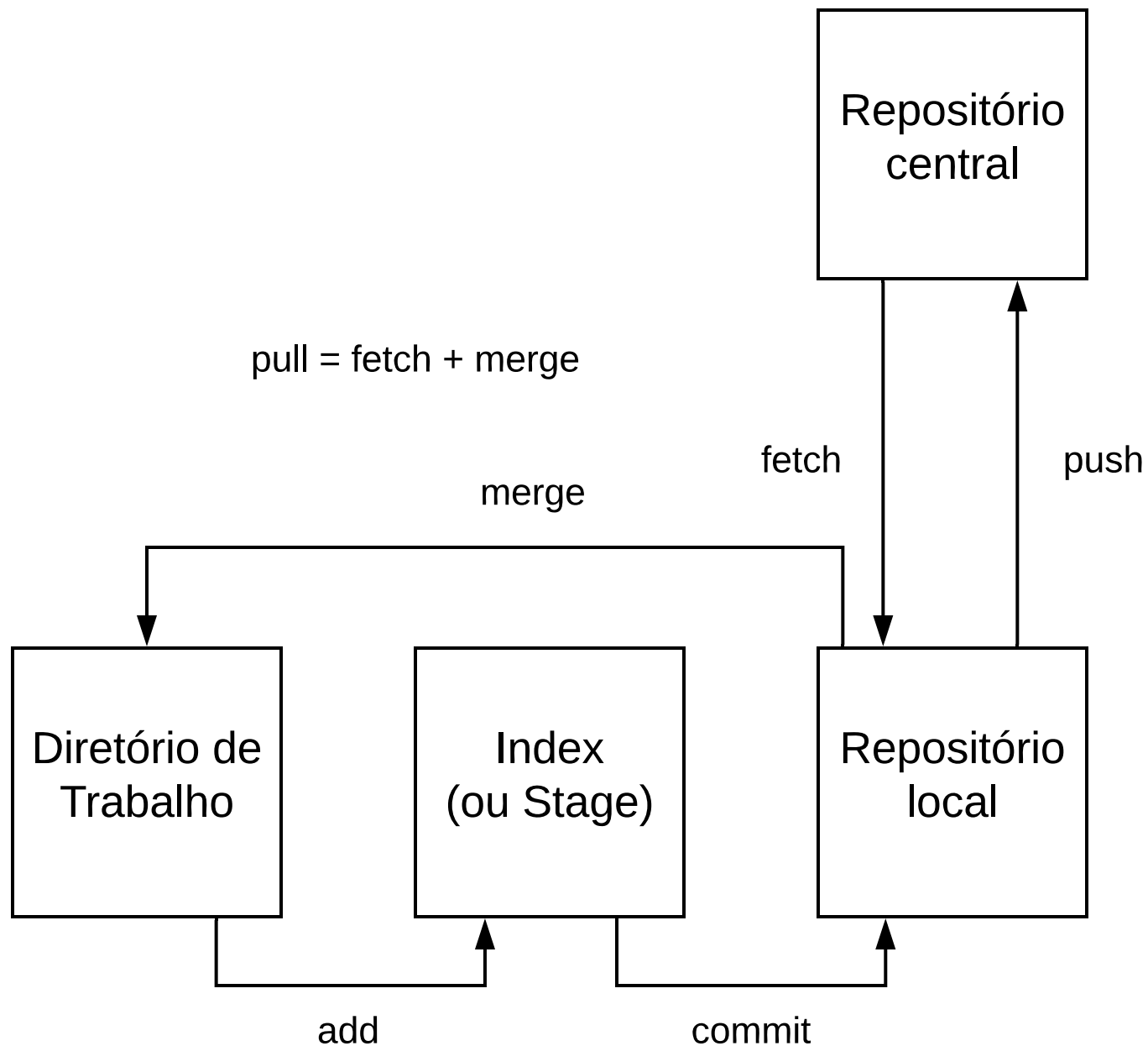
```
git add 'arquivo'
```

```
git commit -m "msg"
```



push e pull

- O comando **push** copia os commits mais recentes do repositório local para o repositório remoto.
-
- Já o comando **pull** faz o contrário, atualiza seu repositório local de acordo com o repositório remoto atual.



Glossário Git

- **Branch** - ramificação
- **Tag** - é uma marcação que aponta para um momento específico no histórico do Git.

Lista de comandos Git

[Link](#)

Resumindo...

Comandos salvar alterações

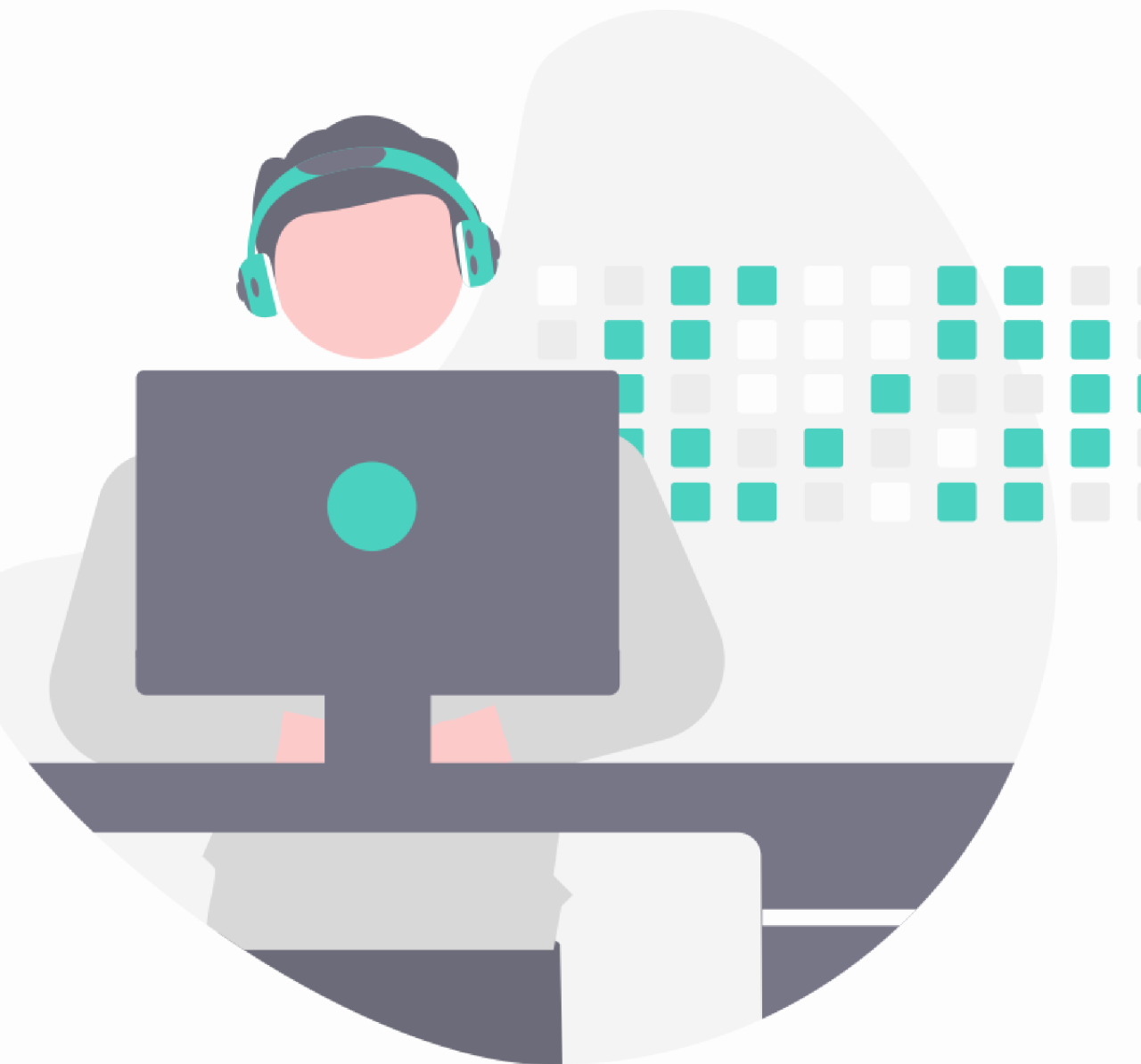
1. `git add .` (adiciona os arquivos alterados na área de stage)
2. `git commit -m "descrever alterações"` (salva as alterações no repositório git local)
3. `git push` (envia todas as alterações do repositório git local para o repositório remoto)

Comandos para receber alterações

- `git pull` - atualiza o repositório local de acordo com o repositório remoto atual.

Objetivos de aprendizagem

- Conhecer o Git ✓
- Entender a necessidade e vantagens do Git ✓
- Conhecer o GitHub ✓
- Aprender os comandos básicos do Git ✓
- Criar um projeto no GitHub ✓



Hora do aluno

Criar um repositório no
GitHub e fazer pelo menos
dois commits

Saibas mais

1. <https://git-scm.com/book/pt-br/v2>
2. <https://www.atlassian.com/br/git/tutorials/what-is-git>
3. <https://blog.rocketseat.com.br/iniciando-com-git-github/>
4. <https://guides.github.com/activities/hello-world/>
5. <https://engsoftmoderna.info/capAp.html> (As figuras descritivas dos comandos acima foram retiradas deste link)

@jesielviana

jesielviana.com