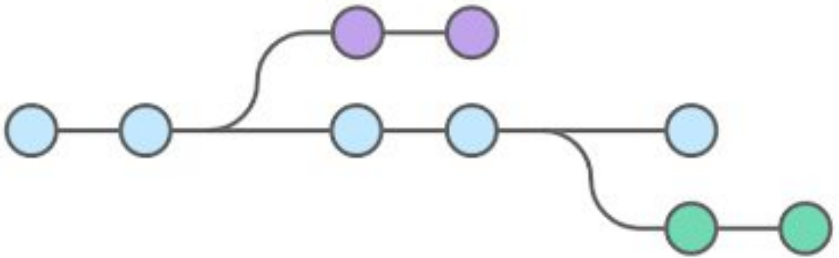
OCEAN

git

SAMSUNG

O que é GIT ?

Git é um sistema de controle de versões distribuído, usado principalmente no desenvolvimento de software, mas pode ser usado para registrar o histórico de edições de qualquer tipo de arquivo.



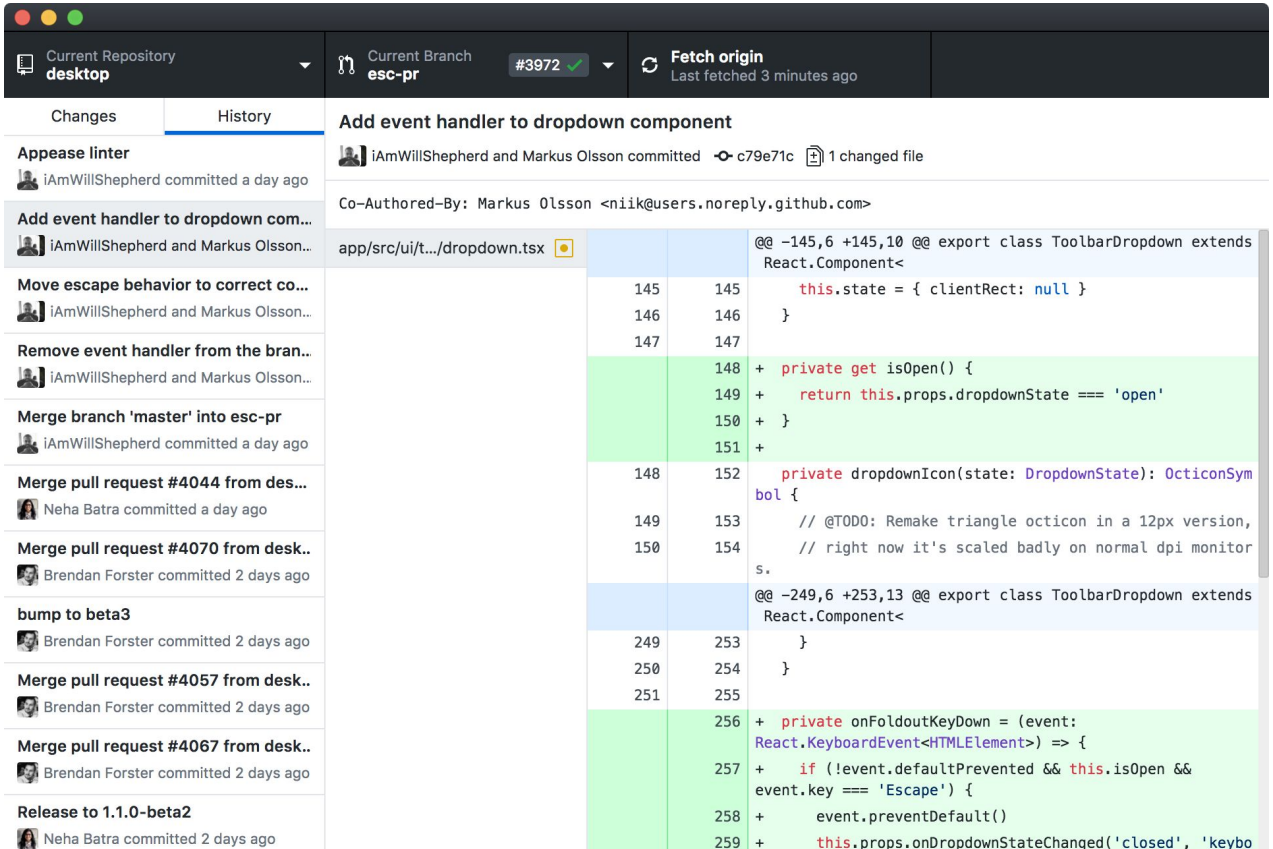
Quem criou ?

Em 2005, Linus Torvalds (o homem conhecido por criar o kernel Linux OS) desenvolveu o GIT e, desde então, tem sido ativamente mantido por Junio Hamano

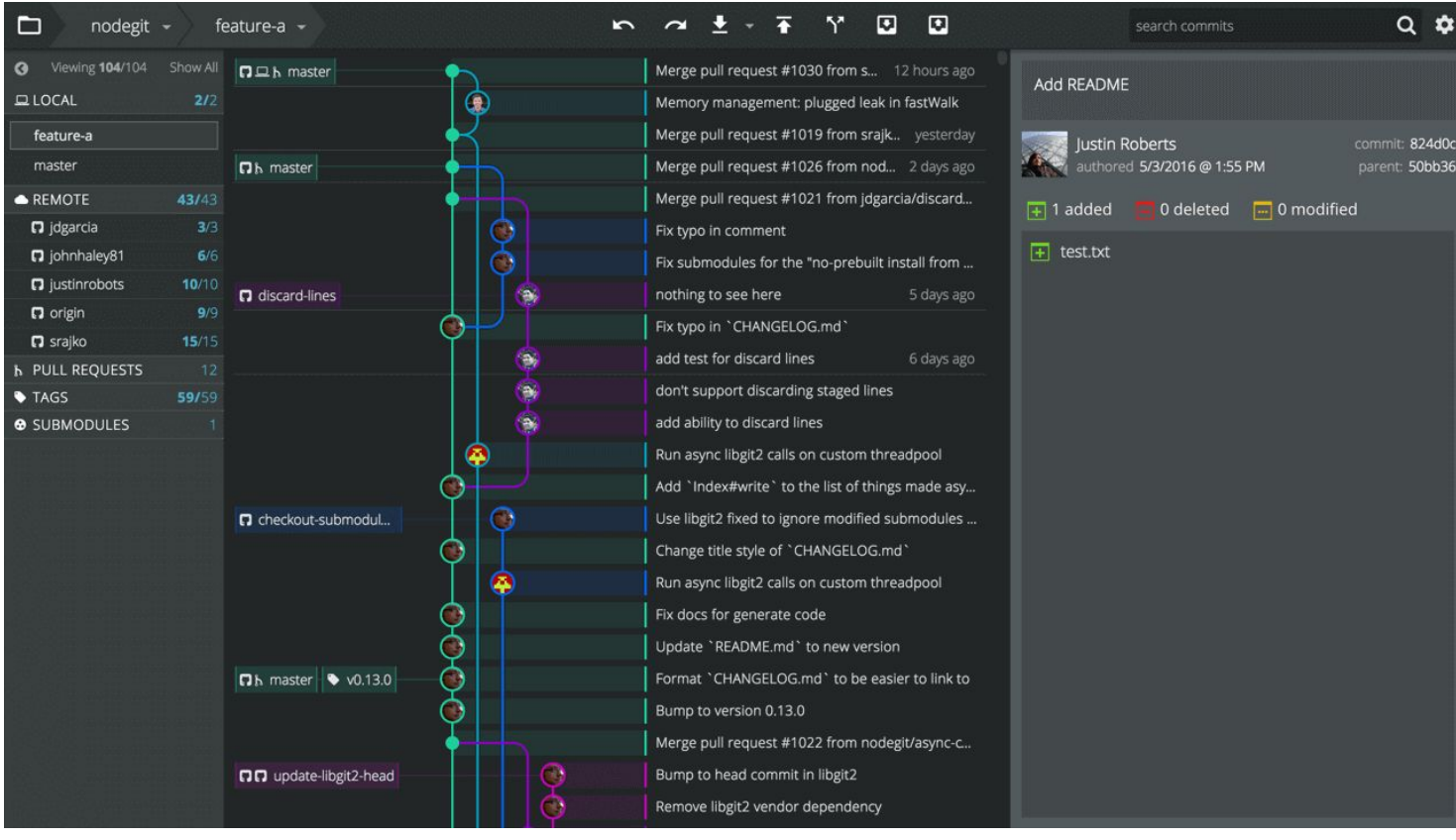
Modos de utilizar

- Interface gráfica
- Linha de comandos

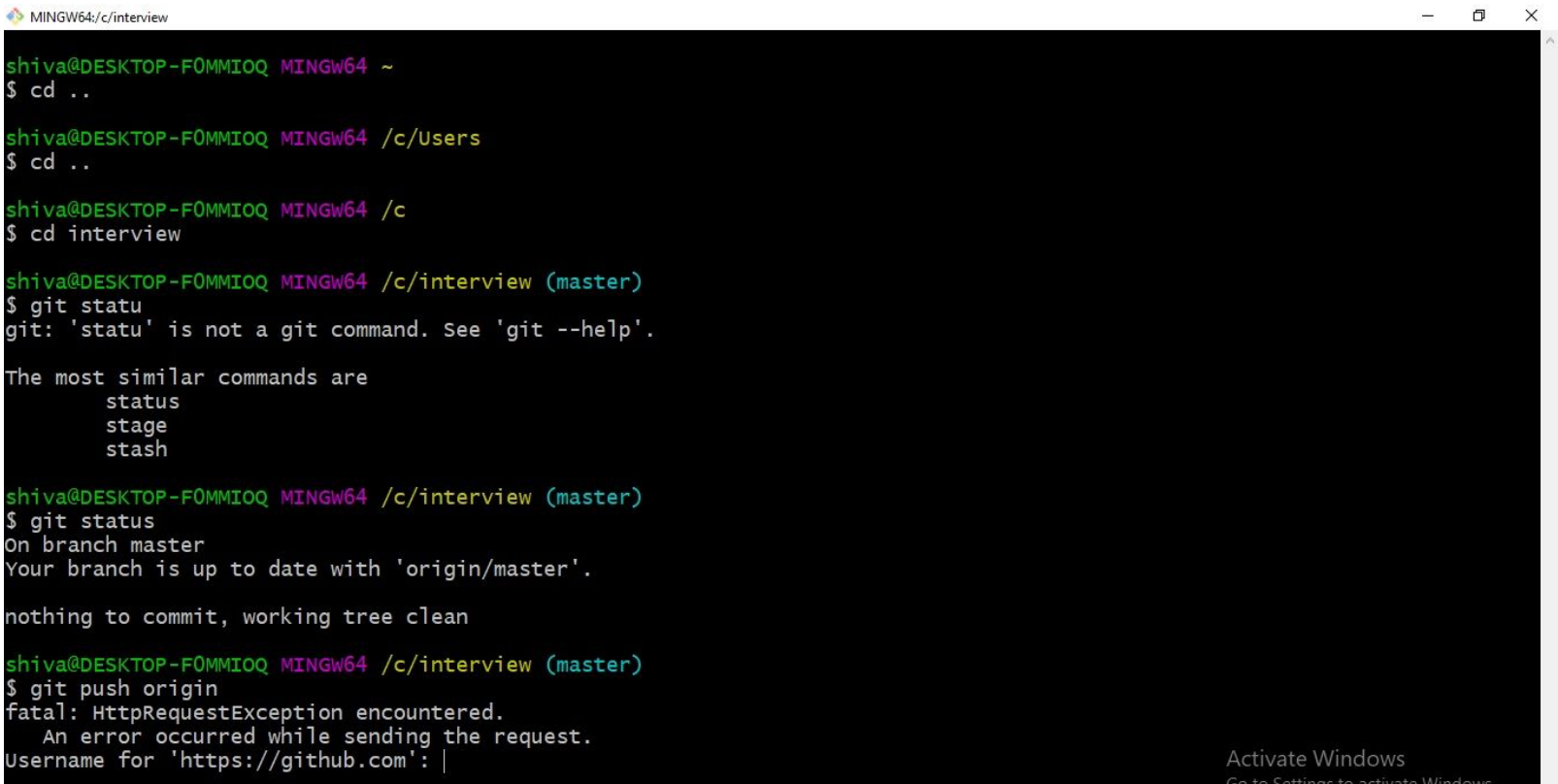
Interface Gráfica



Interface Gráfica



Linha de comando



Vamos utilizar linha de comando!

Faça download do git bash do github

- Download github Bash - https://gitforwindows.org/

Comandos básicos

- **cd** - abrir uma pasta
- **git init** - transforma a pasta em um projeto git
- **git add** - rastreia os arquivos que serão comitados
- **git commit** - salva todas as alterações feitas até aquele momento
- **git push** - envia as modificações para o servidor

Comando “cd”

- cd /pasta_do_projeto
- cd ..
- cd diretorio/subdoretorio/projeto

Para configurar o usuário do git

- git config --global user.name "John Smith"
- git config --global user.email "example@email.com"

Git init

Todo comando git deve ser executado dentro de um repositório git.

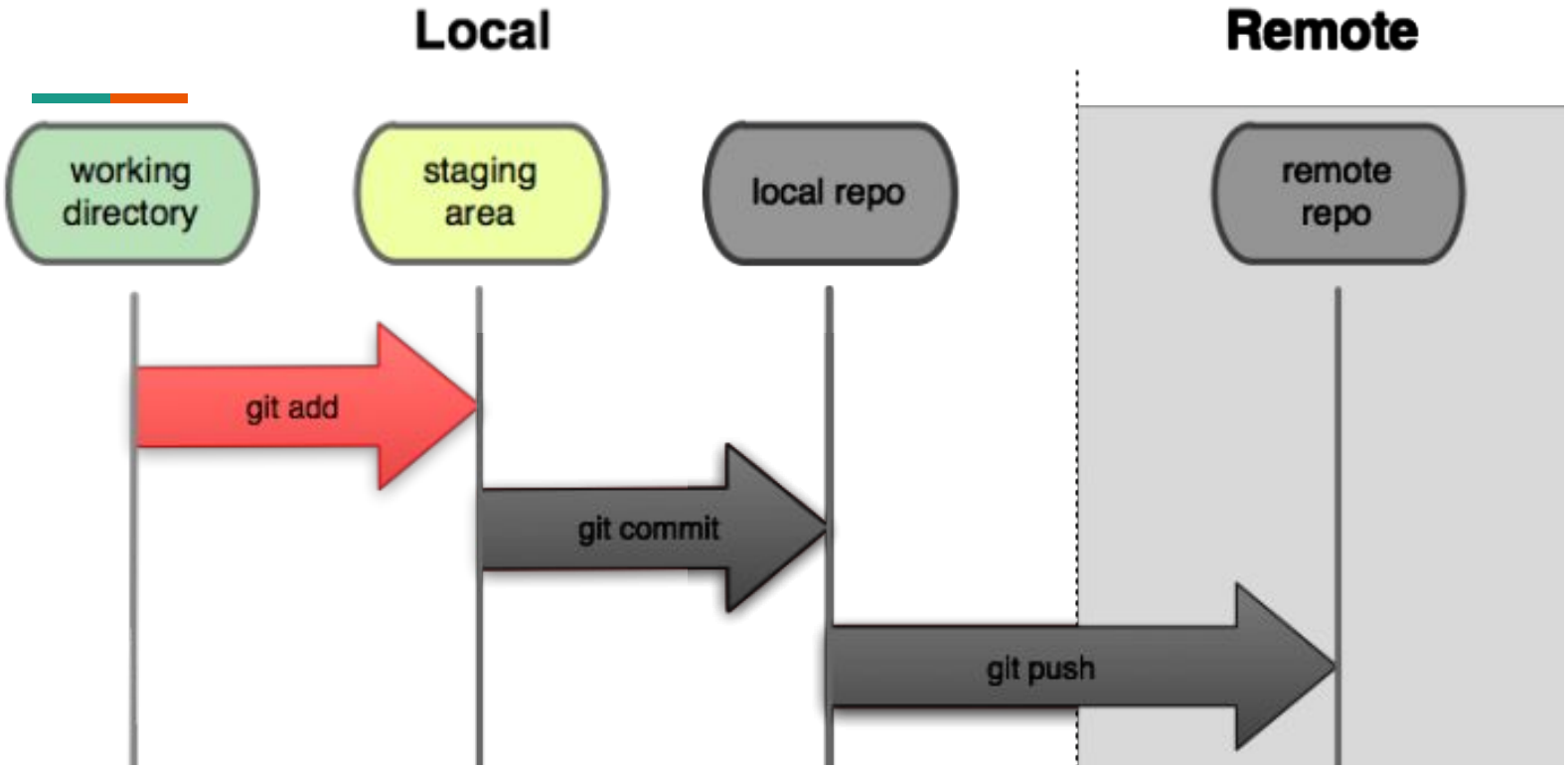
Para transformar um diretório comum em um repositório git, temos que utilizar o comando “**git init**”

Git add

Rastreia os arquivos que deverão ser comitados

git add <file_name>

- git add teste.java
- git add *.java
- git add .

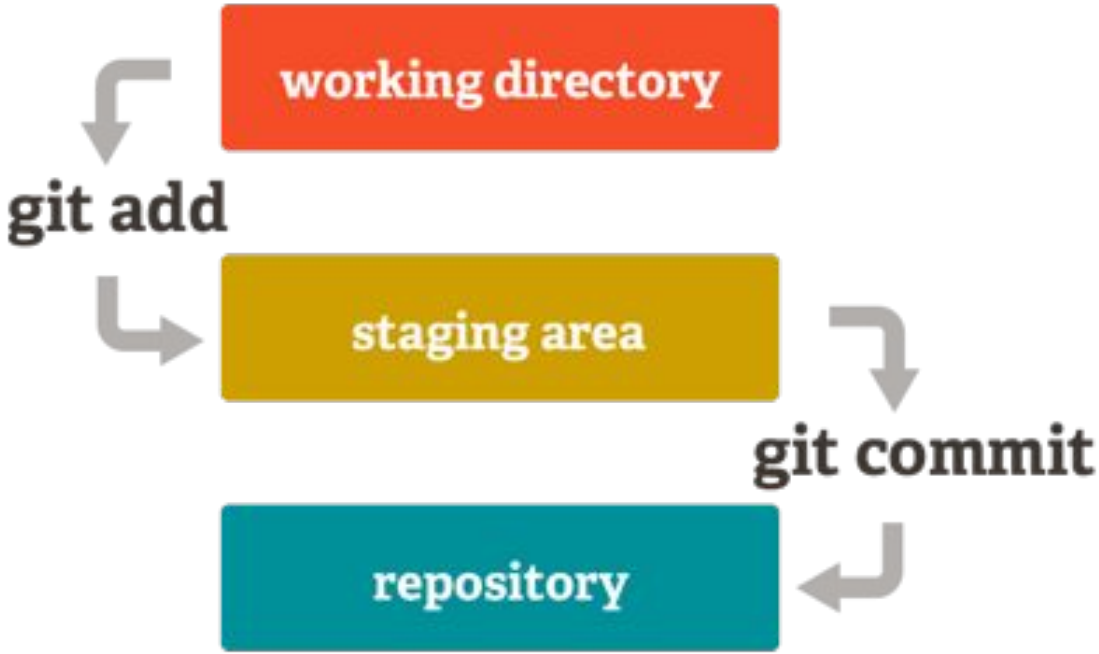

Local
Remote
working directory
staging area
local repo
remote repo
git add
git commit
git push

Git commit

Salva todos os arquivos mapeados até aquele momento.

Exemplos

- git commit -m “Primeiro commit”
- git commit -m “Criação da classe calculadora”
- git commit -m “Implementação dos testes para calculadora”

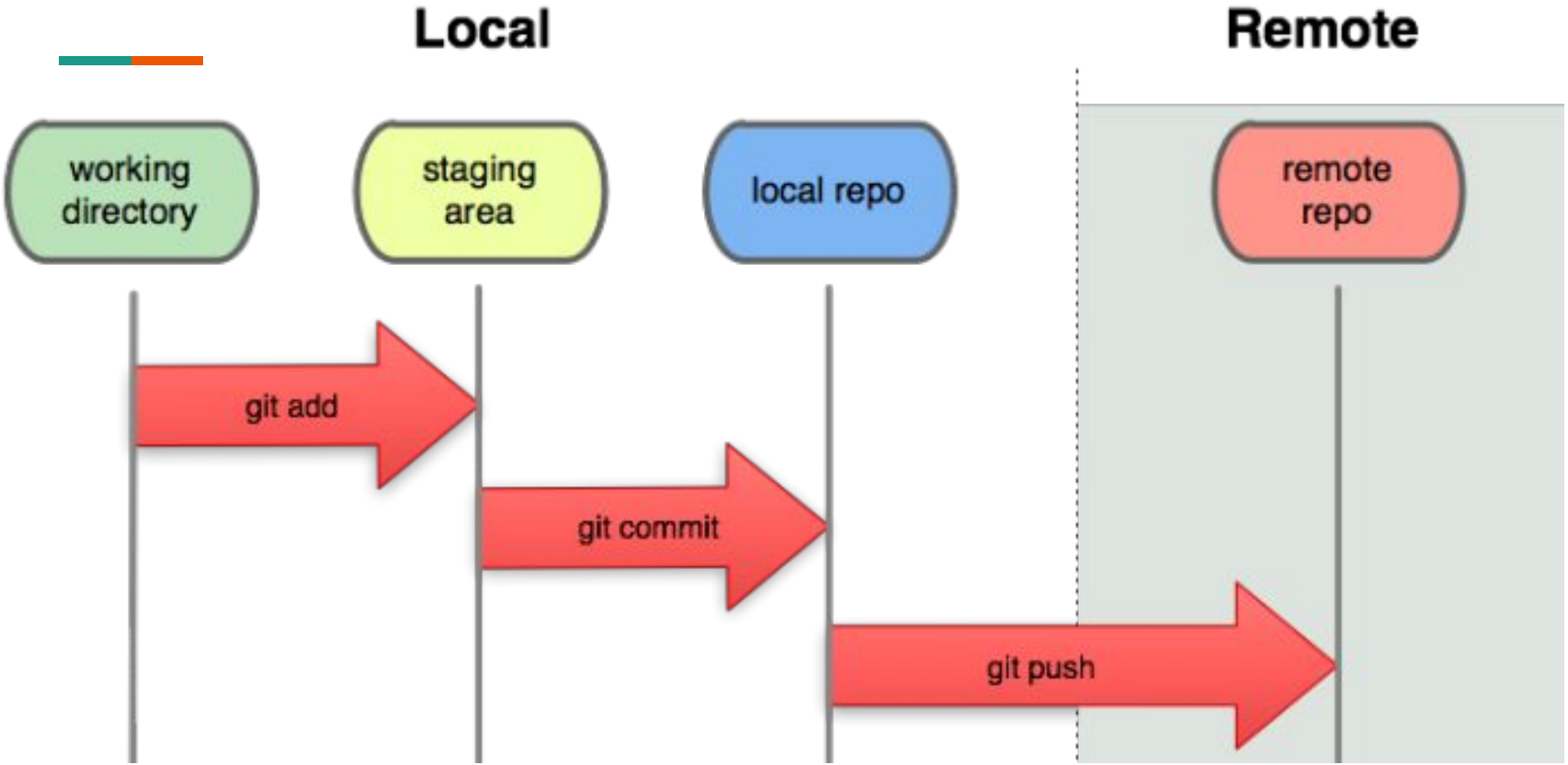

working directory
git add
staging area
git commit
repository

Git push

Salva todas as modificações no repositório remoto

Exemplos

- git push origin master


Local
Remote
working directory
staging area
local repo
remote repo
git add
git commit
git push

Repositório remoto

GitLab

Criando conta no github

- Criar uma conta no site https://github.com/
- Criar um repositório

Repositório remoto

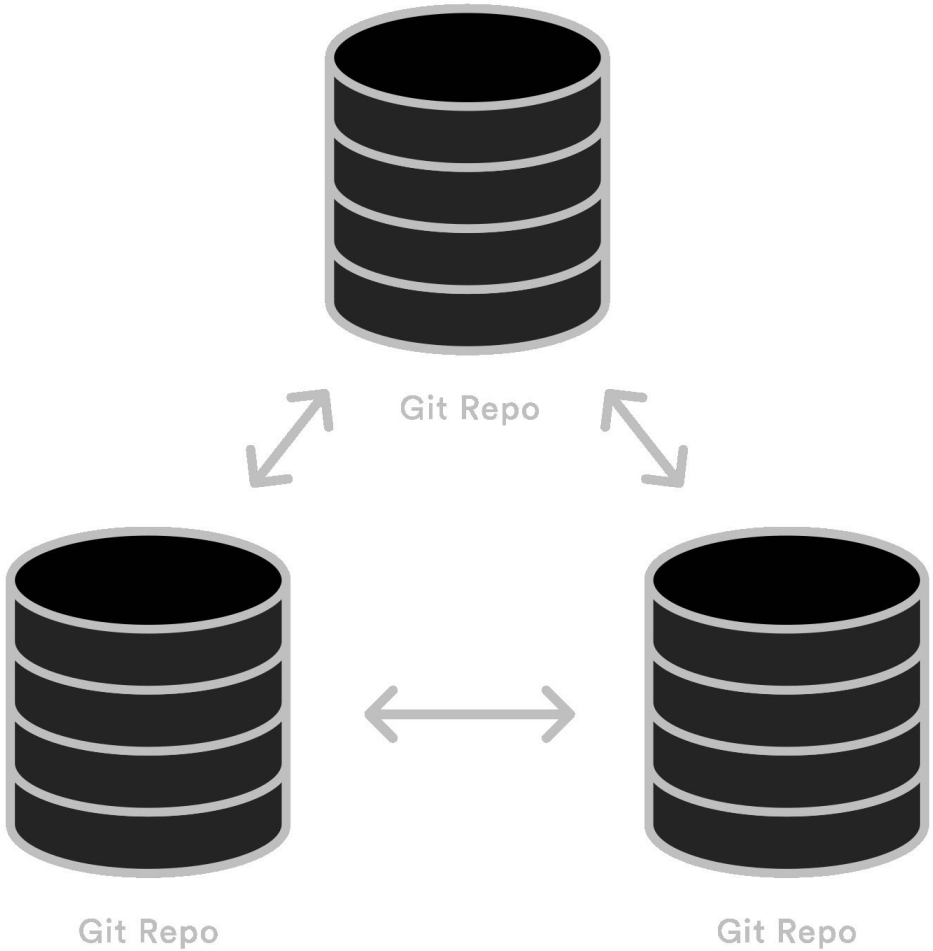
Para clonar o repositório remoto para o seu pc:

- `git clone user.name@host:/path/to/remote/repository`

Para vincular um projeto git já existente com o repositório remoto:

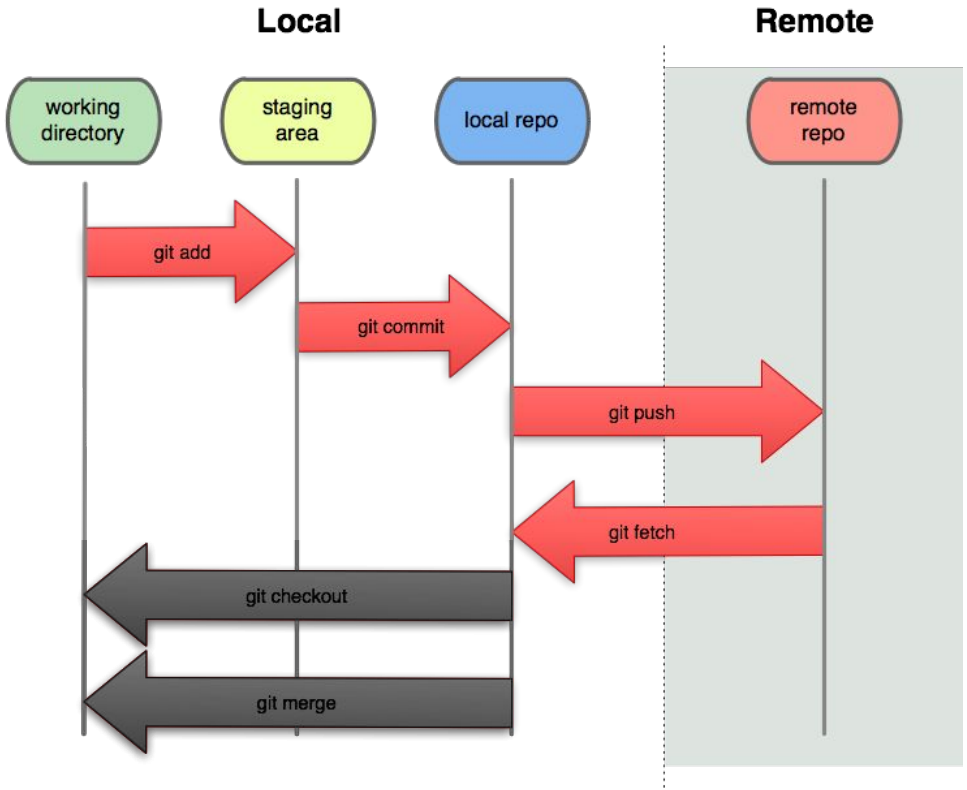
- `git remote add origin  https://github.com/projeto.git`

Git clone



Git fetch

Busque ramificações de um ou mais outros repositórios, juntamente com os objetos necessários para completar seus históricos. As ramificações de rastreamento remoto são atualizadas.



Git checkout

Para alternar entre as versões existentes

git checkout <commit>

- **git checkout** 0ee5959be455eddbd90f5217be75ac79a3e60954

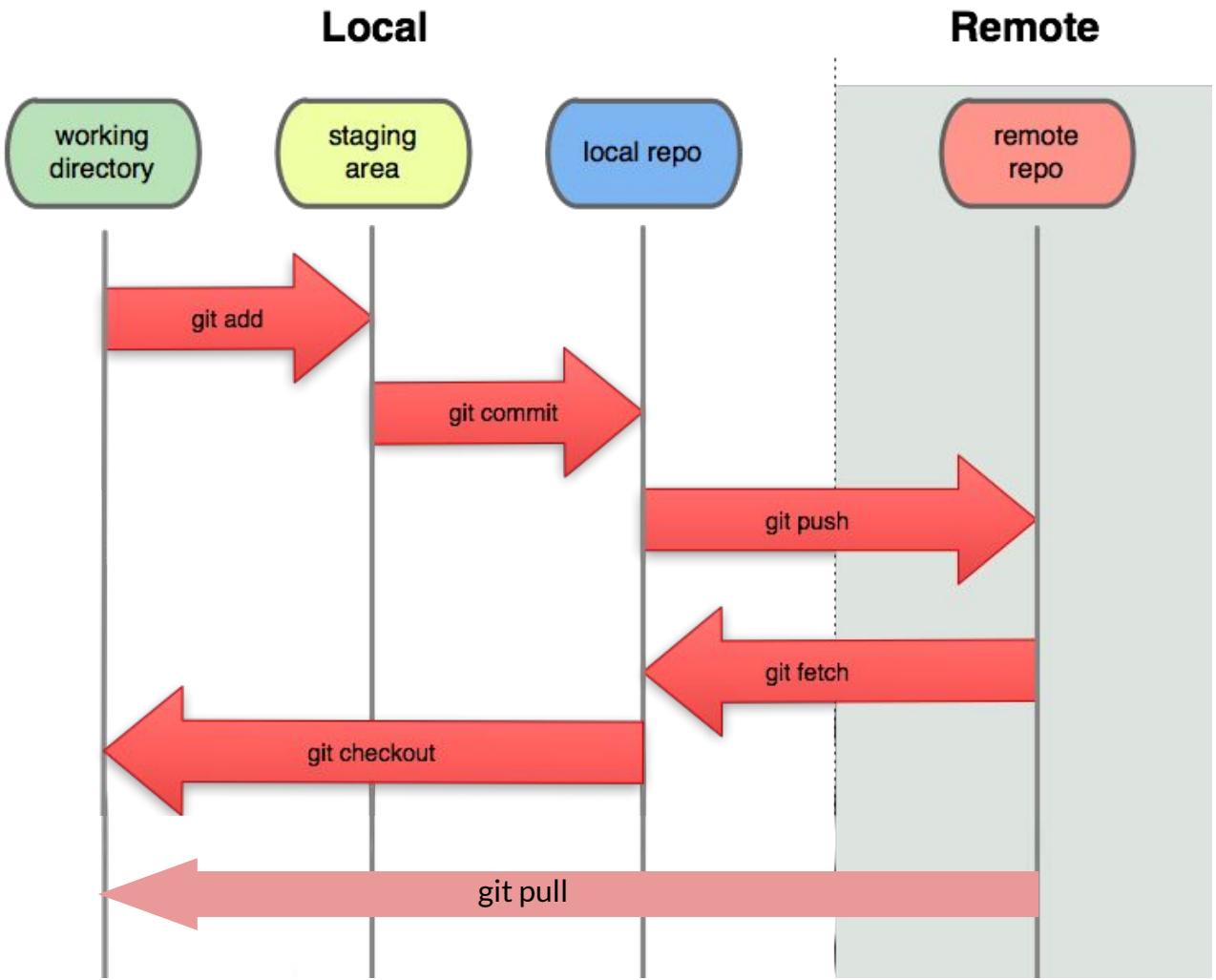
Git pull

Traz as mudanças do repositório remoto para o local.

Exemplos:

- git pull

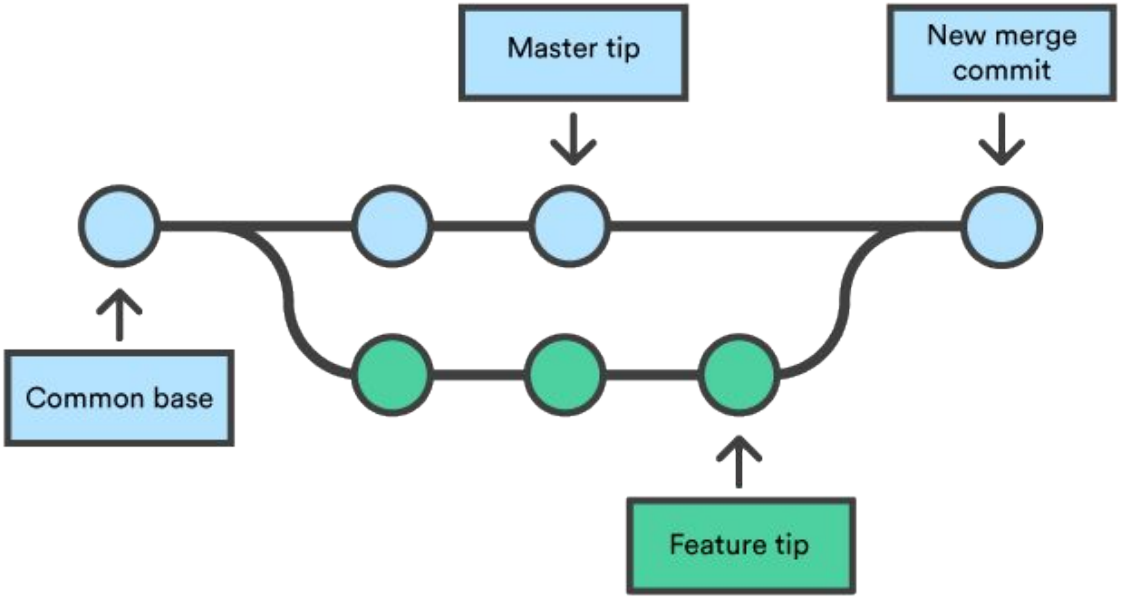
Git pull


Local
Remote
working directory
staging area
local repo
remote repo
git add
git commit
git push
git fetch
git checkout
git pull

Branches

Outra característica do GIT é sua capacidade de permitir que desenvolvedores e gerentes de projeto criem vários ramos independentes dentro de um único projeto.

O objetivo principal de um ramo é desenvolver recursos, mantendo-os isolados uns dos outros. O ramo padrão em qualquer projeto é sempre o ramo mestre.


Master tip
New merge commit
Common base
Feature tip

Criando uma nova branch

Um novo ramo pode ser criado usando o seguinte comando:

```
git checkout -b nome_da_branch
```

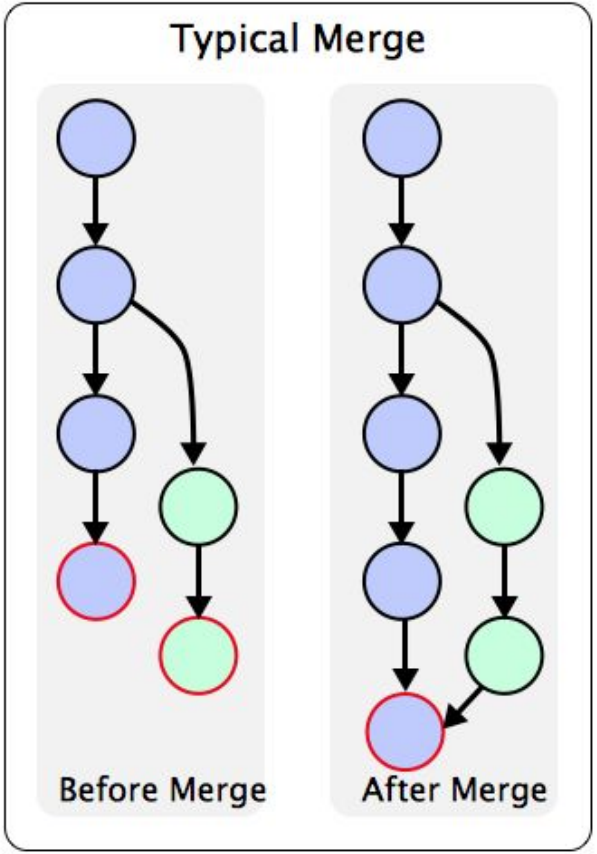
Enviando branch para o repositório remoto

- git push origin featureName
- git push origin master

Merge da Branch na Master

O merge serve para juntar duas branches

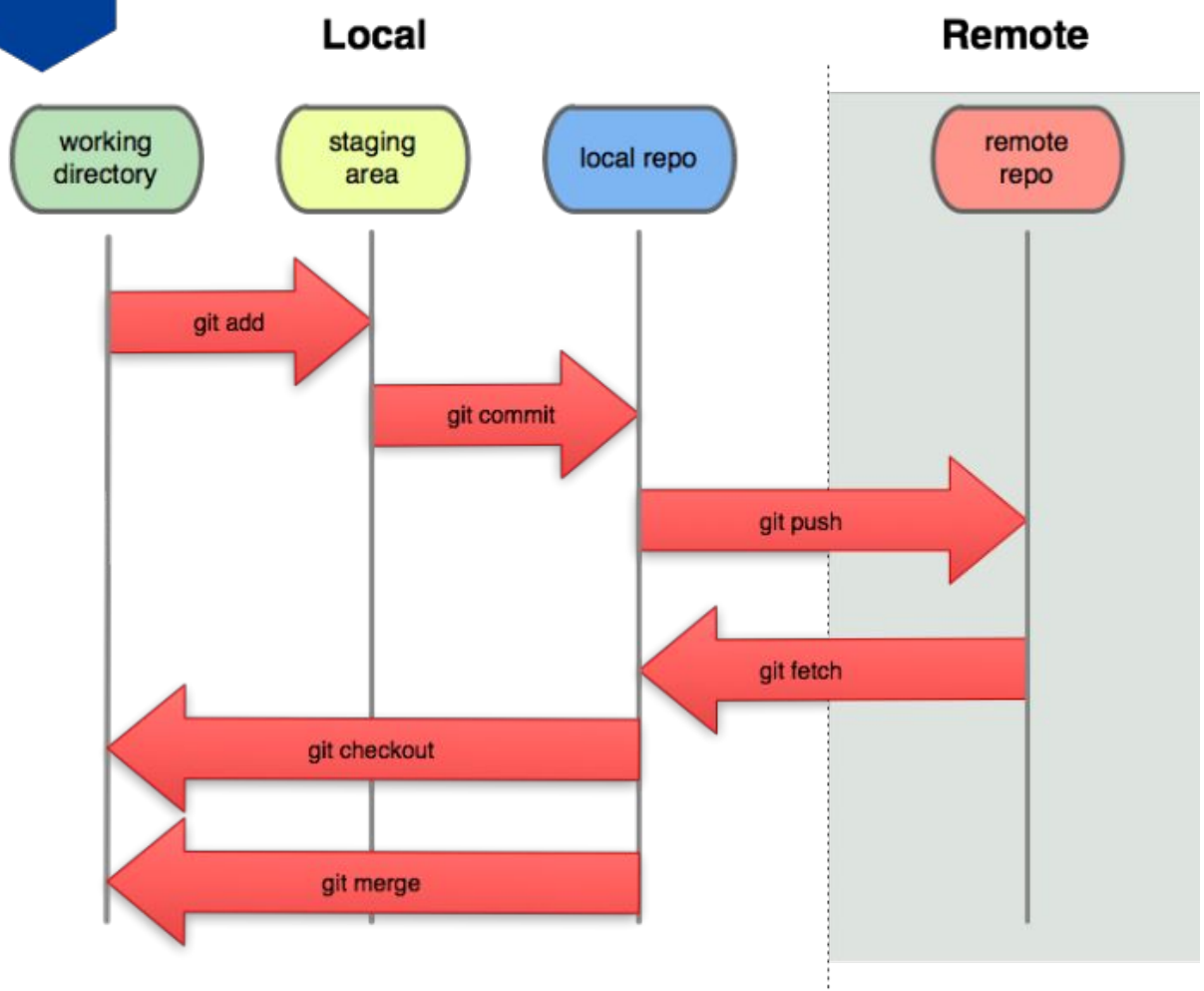
git merge <nome_da_branch>



Git checkout

Para alternar entre os commits ou entre as branches utilizamos o comando “git checkout”

- git checkout nome_da_branch


Local
Remote
working directory
staging area
local repo
remote repo
git add
git commit
git push
git fetch
git checkout
git merge

Tarefa 1

- Crie um repositório online no github
- Clone esse repositório na sua máquina
- Adicione um arquivo de texto
- Faça um commit
- Adicione um arquivo de imagem nesse diretório
- Faça outro commit
- Faça um push para salvar as modificações no repositório.

Tarefa 2

- Crie uma nova branch nesse repositório
- Adicione três imagens no diretório
- Faça um commit
- Suba as modificações para a nova branch
- Faça o merge da master com essa nova branch


That's all Folks!