



**GITHUB:**

<https://github.com/Cristianomeneses2008/git>

**Palestra: Cristiano Meneses - 2023**

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 (Exemplo: alguns livros digitais são disponibilizados no GitHub e escrito aos poucos publicamente). O Git foi inicialmente projetado e desenvolvido por Linus Torvalds para o desenvolvimento do kernel Linux, mas foi adotado por muitos outros projetos.

Cada diretório de trabalho do Git é um repositório com um histórico completo e habilidade total de acompanhamento das revisões, não dependente de acesso a uma rede ou a um servidor central. O Git também facilita a reprodutibilidade científica em uma ampla gama de disciplinas, da ecologia à bioinformática, arqueologia à zoologia.[4]

O Git é um software livre, distribuído sob os termos da versão 2 da GNU General Public License. Sua manutenção é atualmente supervisionada por Junio Hamano.

No **Scrum** é essencial ter um produto liberável ao final de cada Sprint. Para poder lançar com tanta frequência você precisa ter uma boa estratégia de ramificação. Felizmente existem ferramentas que permitem gerenciar o software da melhor maneira possível tanto em termos de ferramentas quanto de processos.

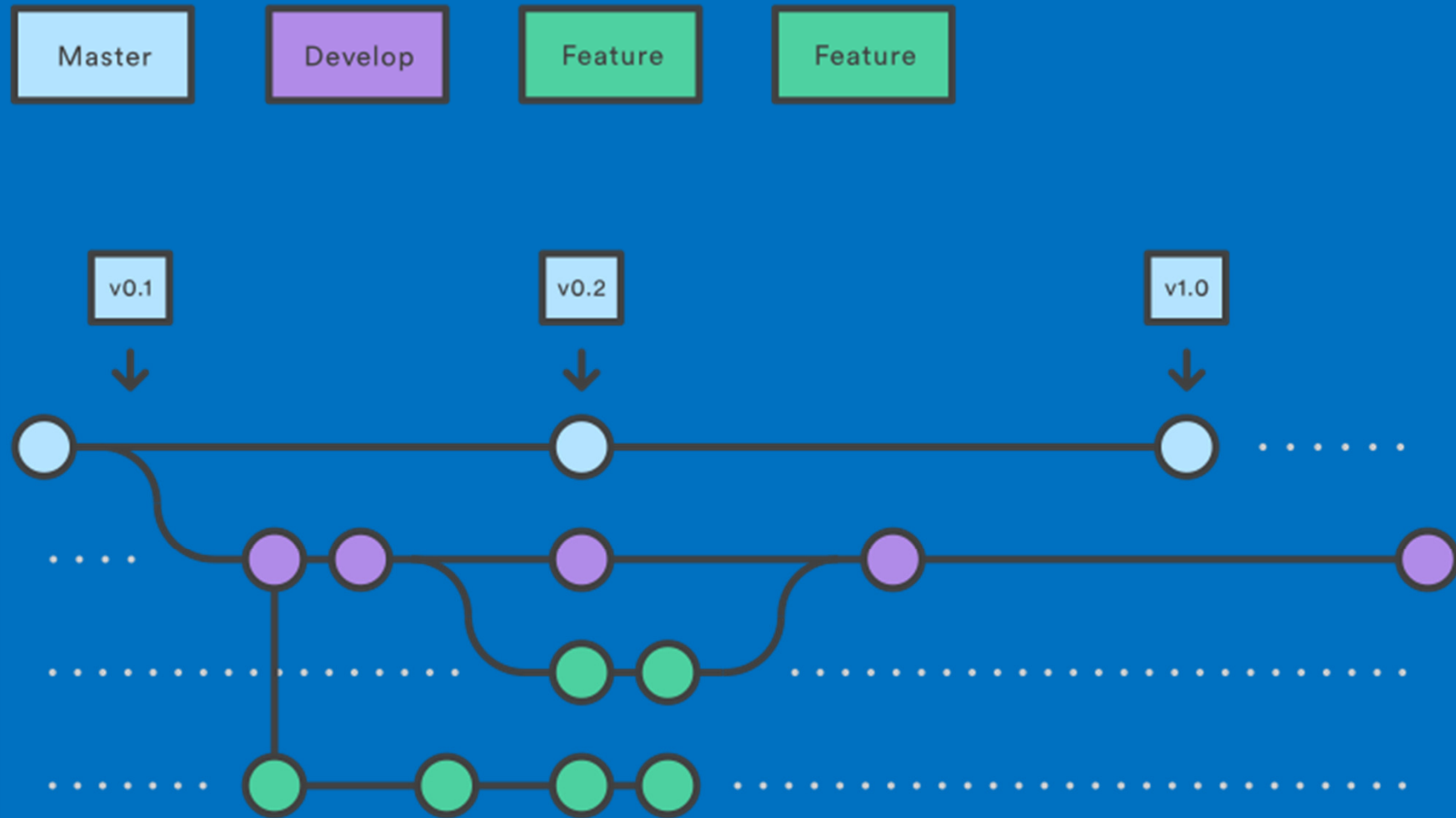
**GitFlow** é um conjunto de diretrizes que sugere a melhor maneira de gerenciar filiais com Git; Seguindo as orientações é possível ter sempre versões de software estáveis que incluam um conjunto de recursos conhecidos.



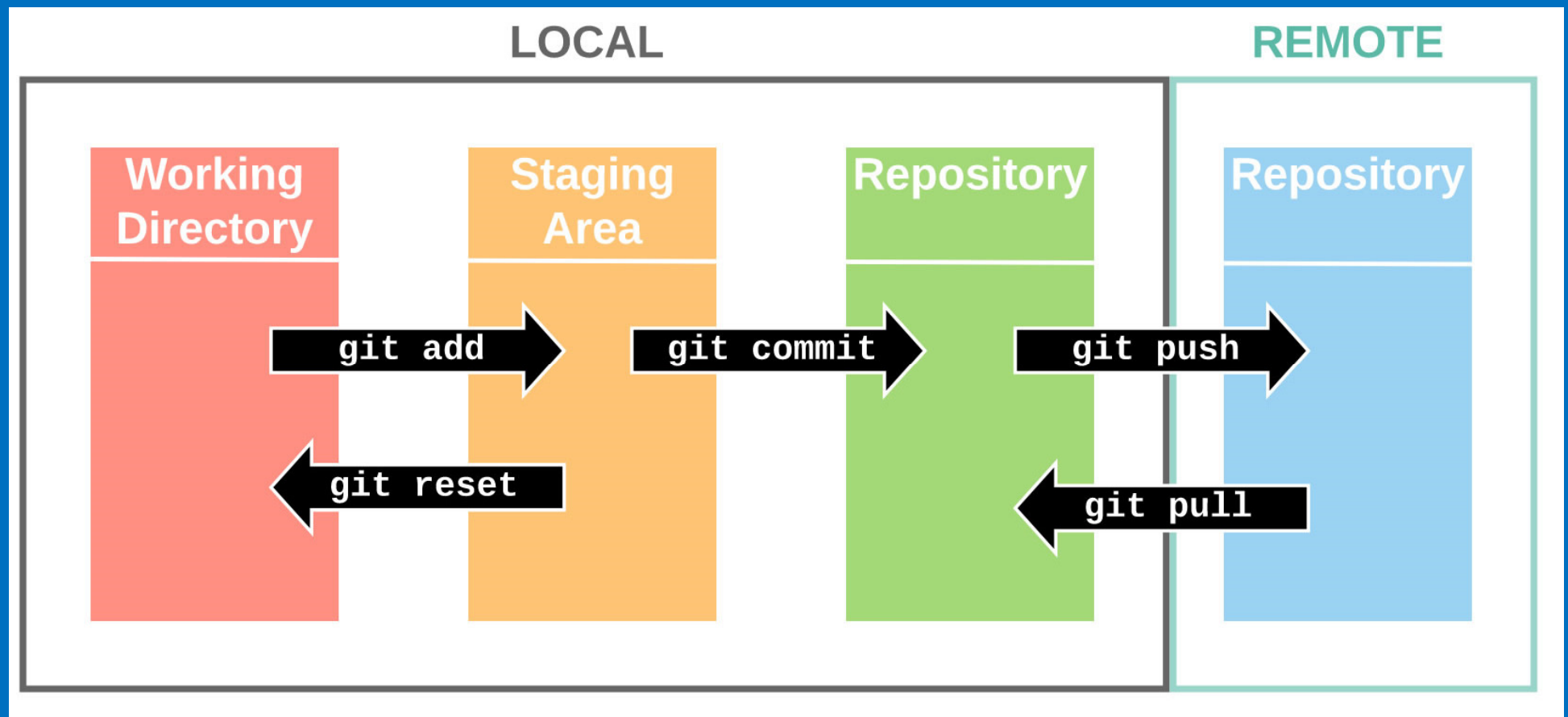
# Desenho



# Gitflow



# Entendendo o fluxo do versionamento.



## Configurando o GIT

Os comandos mostrados neste artigo foram testados no Windows. Uma vez instalado, abra o terminal do Windows e configure o git com os dados de sua conta do Github

Para isso, utilize os comandos **git config**

A primeira coisa a fazer depois de instalar o Git é definir o seu nome de usuário e endereço de e-mail. Isso é importante porque todos os commits no Git utilizam essas informações

```
git config --global user.name "Seu nome de usuário"
```

```
git config --global user.email seuemail@seuservidor.com
```

Para copiar um repositório do GitHub para a sua máquina, utilize o comando

**Git clone <Endereço remoto do repositório>**

Para transformar a pasta do projeto num repositório do GIT, ou para ativar o repositório existente, dentro da pasta do projeto, utilize o comando:

**Git init.**

## **git add nome\_do\_arquivo**

Adiciona o arquivo na área de index. Para adicionar todos os arquivos que sofreram alteração (ou para copiar todos os arquivos caso seja a primeira vez), utilize **git add** .

## **git status**

Com este comando você consegue ver quais arquivos estão fora do controle, quais foram modificados e estão esperando por uma descrição de modificação. Ele também mostra em qual **branch** você se encontra no momento.

## **git reset HEAD nome\_do\_arquivo**

Volta ao estágio anterior do adicionamento. Se você informar **git reset** sem nenhum parâmetro após fazer o git add ele desfaz a operação

Depois de adicionar os arquivos, chegou a hora de guardar. Para isso, executamos o comando commit.

## **Filtros**

Para filtrar commits pelo nome do autor, podemos fazer desta forma:

**git log --autor=nome-autor**

Se for necessário filtrar por data, existem duas formas.

**git log --after=" YYYY-MM-DD"**

**git log --before=" YYYY-MM-DD"**

**git push - u origin --all**

Este comando copia todos os branches do repositório local e envia para o repositório remote



## **git log**

Mostra todos os commits que você fez

## **git branch**

Exemplos de como trabalhar com branches

Crie um novo branch chamado “develop” e selecione-o usando

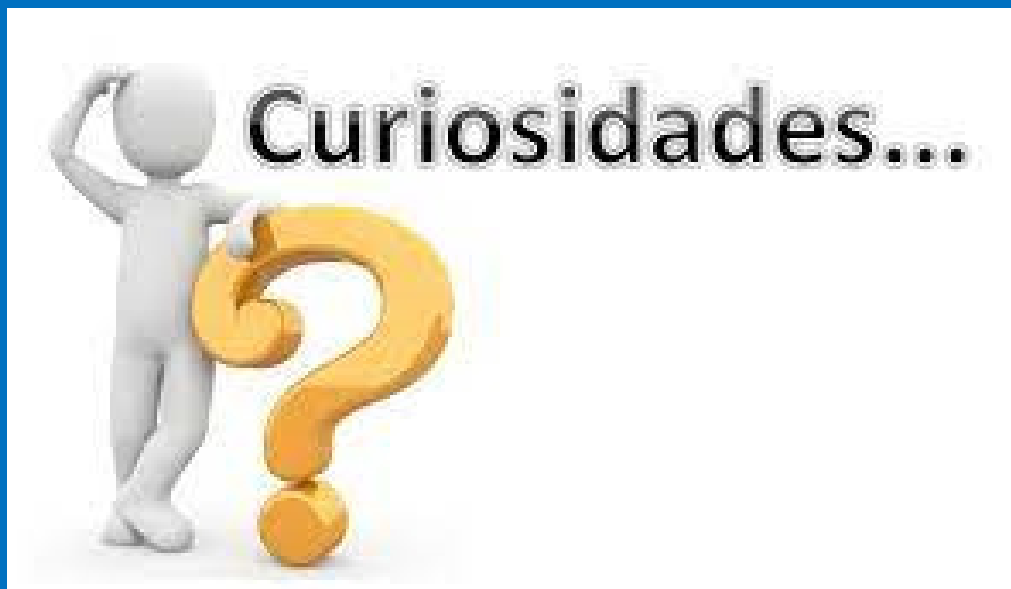
**git checkout –b develop**

Retorne para o branch master usando

**git checkout master**

Para mesclar alterações de um branch com o branch que estiver ativo, use

**git merge nome\_do\_branch**



**Instalações para facilitar:**  
Visual Studio Code  
TortoiseGit

# FIM

## Fontes de Pesquisa:

<https://git-scm.com/>

<https://blog.dbins.com.br/guia-dos-principais-comandos-do-git>

<https://www.docker.com/resources/what-container/>

<https://www.pasqualelangella.com/2020/03/23/scrum-e-la-branching-strategy/>