



ESCOLA DE TECNOLOGIA DA INFORMAÇÃO
INTRODUÇÃO AO GIT

Versionamento de códigos

Geucimar Briatore
geucimar@up.edu.br

Atualizado em 07/2023

O que é e para que serve um sistema de versionamento?

- Um sistema de controle de versões (ou versionamento) é um software que tem a finalidade de gerenciar diferentes versões no desenvolvimento de um documento qualquer. Esses sistemas são comumente utilizados no desenvolvimento de software para controlar as diferentes versões — histórico e desenvolvimento — dos códigos-fontes e também da documentação.

Principais benefícios

- Segurança;
- Trabalho em equipe;
- Controle de histórico e rastreabilidade;
- Marcação e resgate de versões estáveis;
- Ramificação de projeto.

O que é o Git?

- Git é um software de versionamento de linha de comando usado para criar versões de códigos ou documentos e voltar versões anteriores facilmente, assim como analisar o histórico;
- Git não é uma abreviação. Inicialmente o Git foi chamado de *“The stupid content tracker”* e posteriormente alterado para *“Fast, scalable, distributed revision control system”*).
- Antes do Git haviam outros sistemas de versionamento como o CVS (Concurrent Versions System, 1986) e SVN (Apache Subversion, 2000).

Instalação (<https://git-scm.com>)



Search entire site...

About

Documentation

Downloads

GUI Clients

Logos

Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads



macOS



Windows



Linux/Unix

Older [releases](#) are available and the Git source repository is on [GitHub](#).



GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients](#) →

Logos

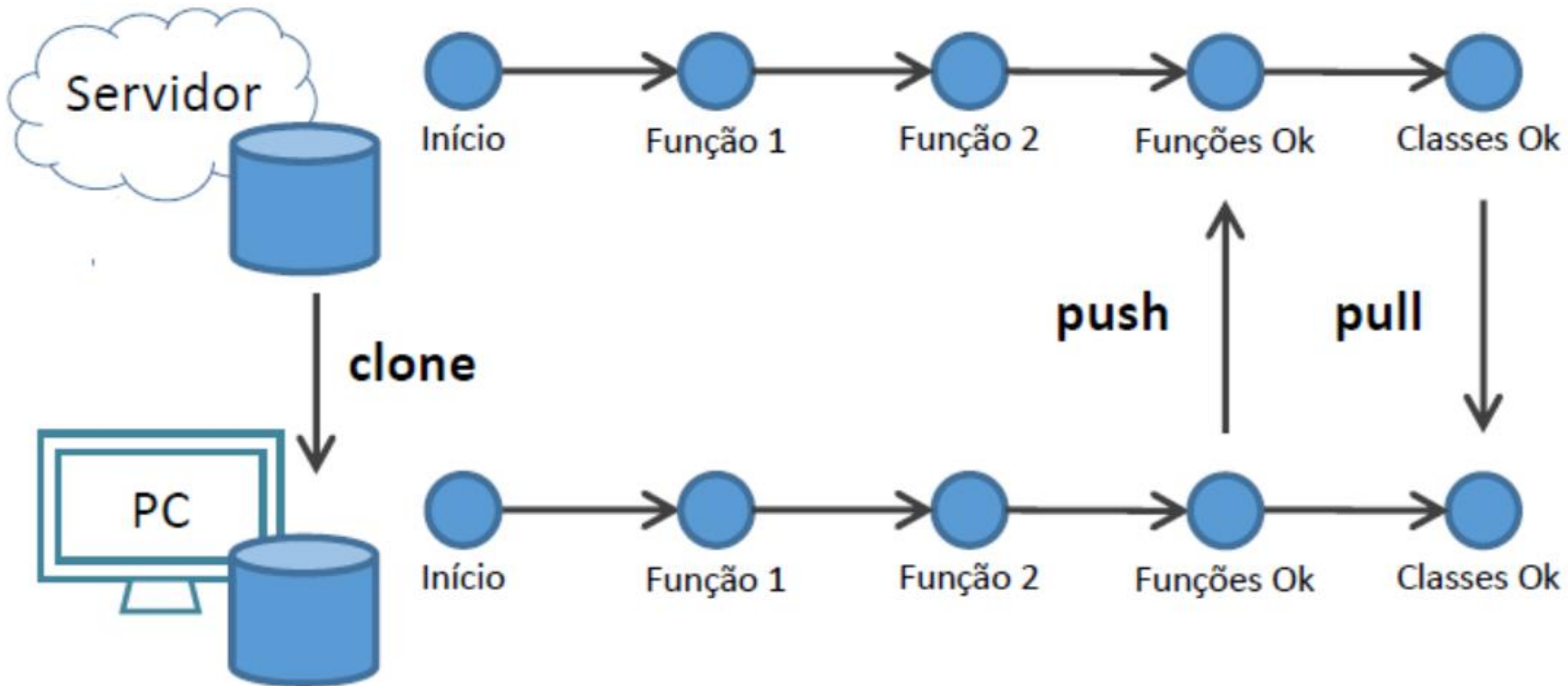
Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos](#) →

Servidores de hospedagem de códigos...



Versionamento de código (Repositório iniciado no servidor)



1. Configuração da credencial no computador local

```
git config --global user.name "nome de usuario"
```

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


2. Clonar repositório remoto, codificar e versionar

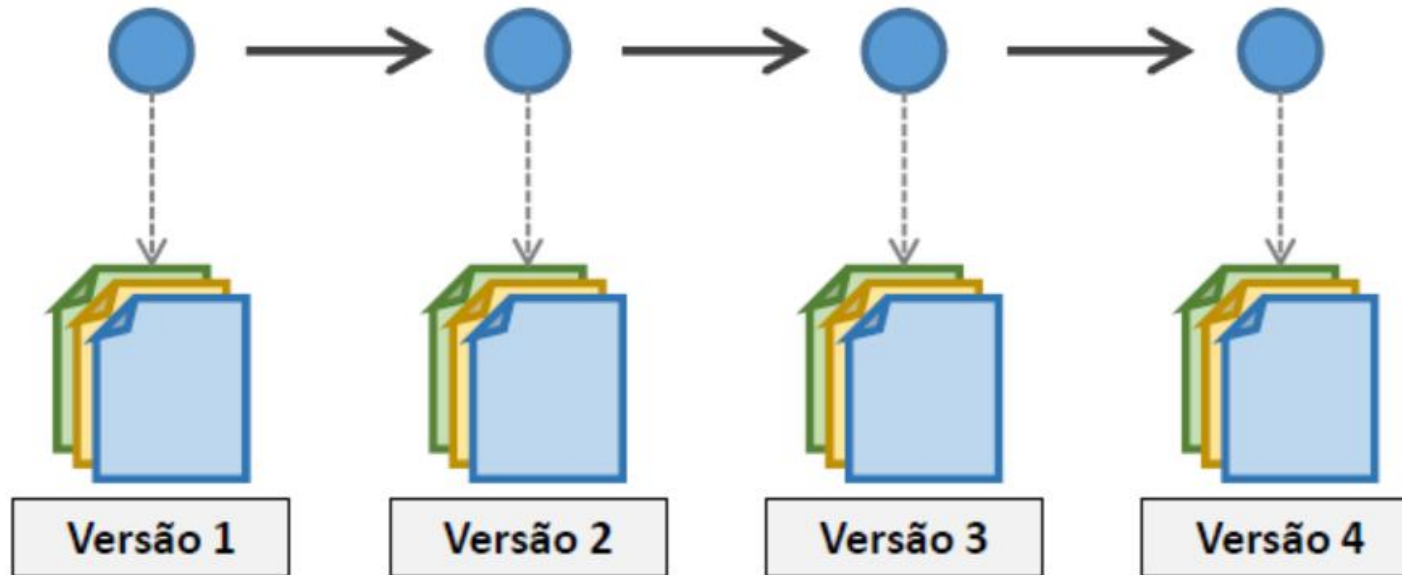
```
git clone "url do repositório remoto"
```

```
git add .
```

```
git commit -m "Descrição da atualização..."
```

```
git push
```

Versões (Snapshots)



3. Codificar, adicionar repositório remoto e versionar

```
git init
```

```
git config --global core.autocrlf true
```

```
git remote add origin https://gitlab.com/xxxx.git
```

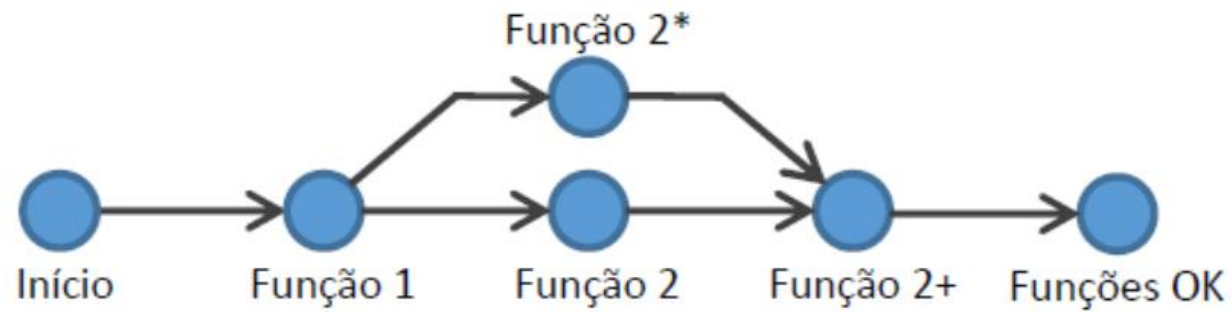
```
git pull origin main --allow-unrelated-histories
```

```
git add .
```

```
git commit -m "Versão inicial"
```

```
git push --set-upstream origin main
```

Ramificações (Branches)



4. Manipulação de Branchs

```
git branch --list //Listar
```

```
git branch <nova branch> //Criar
```

```
git branch -d <branch> //Apagar
```

```
git checkout <branch> //Alternar
```

<https://www.atlassian.com/git/glossary>