

The background of the image is a blurred photograph of a laptop. The screen displays lines of code in a dark-themed editor, with some text highlighted in pink, green, and blue. A portion of the laptop's keyboard is visible at the bottom, with keys illuminated by a yellowish light. Overlaid on this background is a large, bold title in white text on a dark red rectangular background.

CONTROLE DE VERSÃO DE CÓDIGO

Por: Carolina

UM POUCO SOBRE MIM:

- Eu me chamo **Carolina!**
- Sou **estudante de TI**
- Entusiasta de **Software Livre**
- Gosto de buscar **novas experiências**

Contatos: Telegram [@carols0](#) | GitHub
[@mariacarolinass](#)

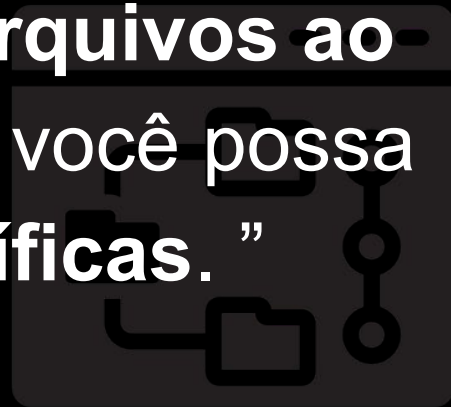


O QUE VAMOS ABORDAR:

- O QUE É CONTROLE DE VERSÃO?
- POR QUE EU DEVERIA USAR UM CONTROLE DE VERSÃO?
- SISTEMA DE VERSÃO DE CÓDIGO (VERSION CONTROL SYSTEM OU VCS)
 - QUAL A IMPORTÂNCIA DE USAR UM SISTEMA DE VERSÃO DE CÓDIGO?
 - TIPOS DE VCS
 - CONTROLE DE VERSÃO LOCAL
 - CONTROLE DE VERSÃO CENTRALIZADO
 - CONTROLE DE VERSÃO DISTRIBUÍDO
- FERRAMENTAS PARA TRABALHAR COM CONTROLE DE VERSÃO DISTRIBUÍDO:
- GIT
- FLUXO DE VERSIONAMENTO DE SOFTWARE (GIT FLOW)
- PLATAFORMAS DE HOSPEDAGEM DE CÓDIGO-FONTE COM CONTROLE DE VERSÃO
 - GITHUB
- HORA DA PRÁTICA
- REFERÊNCIAS

O QUE É CONTROLE DE VERSÃO?

“ O **controle de versão** é um sistema que registra as **mudanças feitas em um arquivo** ou um **conjunto de arquivos ao longo do tempo** de forma que você possa **recuperar versões específicas.** ”



POR QUE EU DEVERIA USAR UM CONTROLE DE VERSÃO?



projetooficial



projetooficial1



projetooficial2



projetooficial-copia



projetooficial-
copia-final



projetooficial-
copia-final-1



projetooficial-
copia-final-2

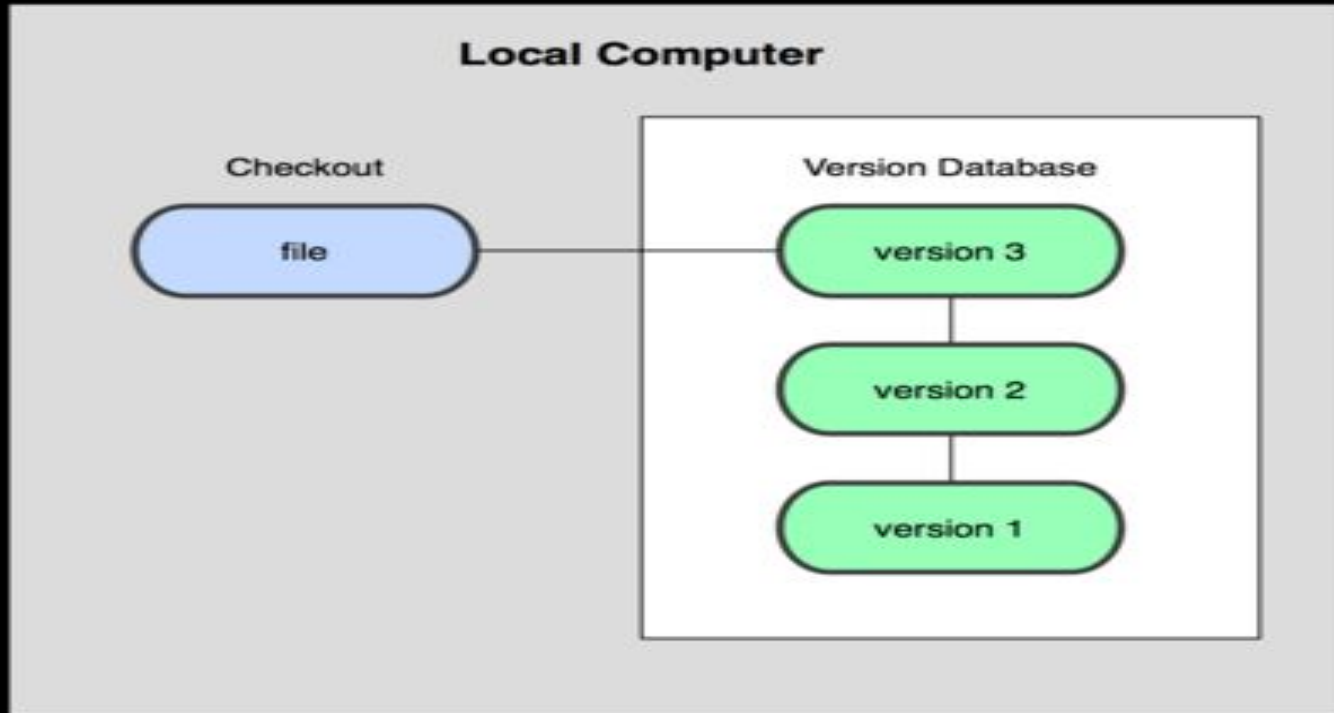
SISTEMA DE VERSÃO DE CÓDIGO **(VERSION CONTROL SYSTEM OU VCS)**

QUAL A IMPORTÂNCIA DE USAR UM SISTEMA DE VERSÃO DE CÓDIGO?

- 1. Voltar no tempo** (Recuperar uma versão específica feita anteriormente)
- 2. Comparar mudanças**
- 3. Ver última pessoa que fez mudanças**

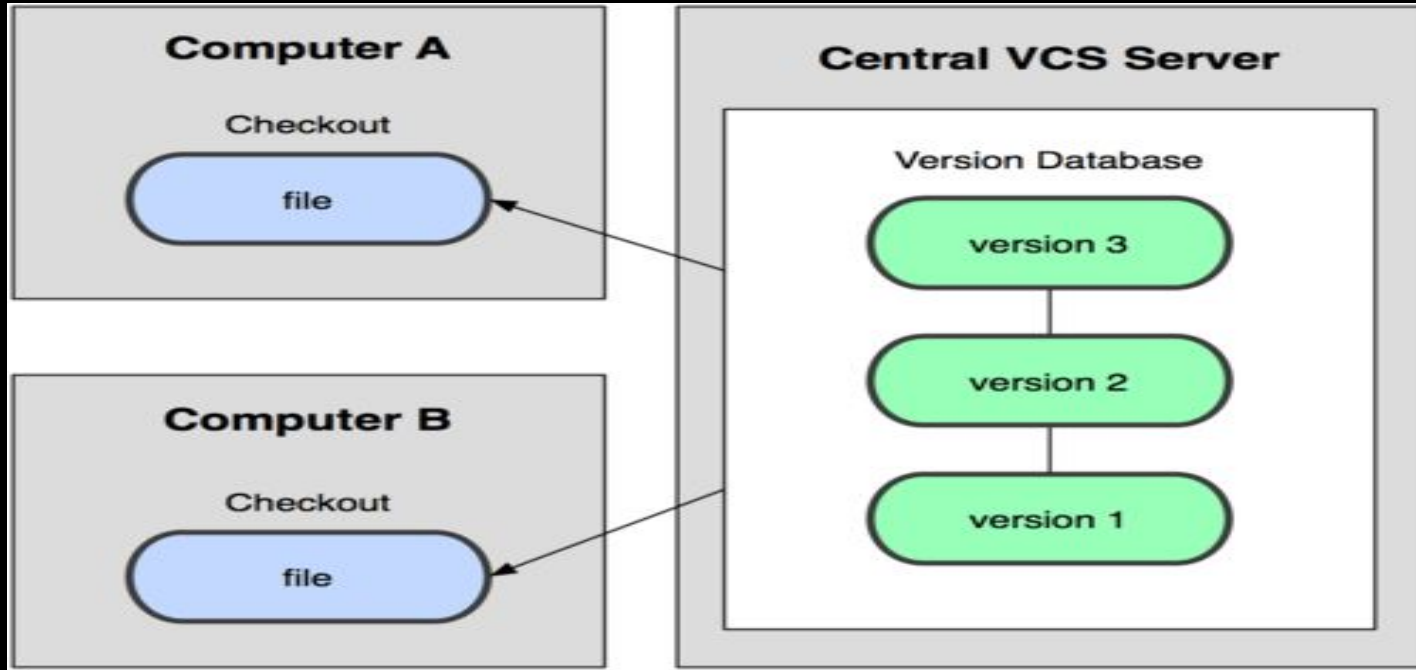
TIPOS DE VCS

1. CONTROLE DE VERSÃO LOCAL



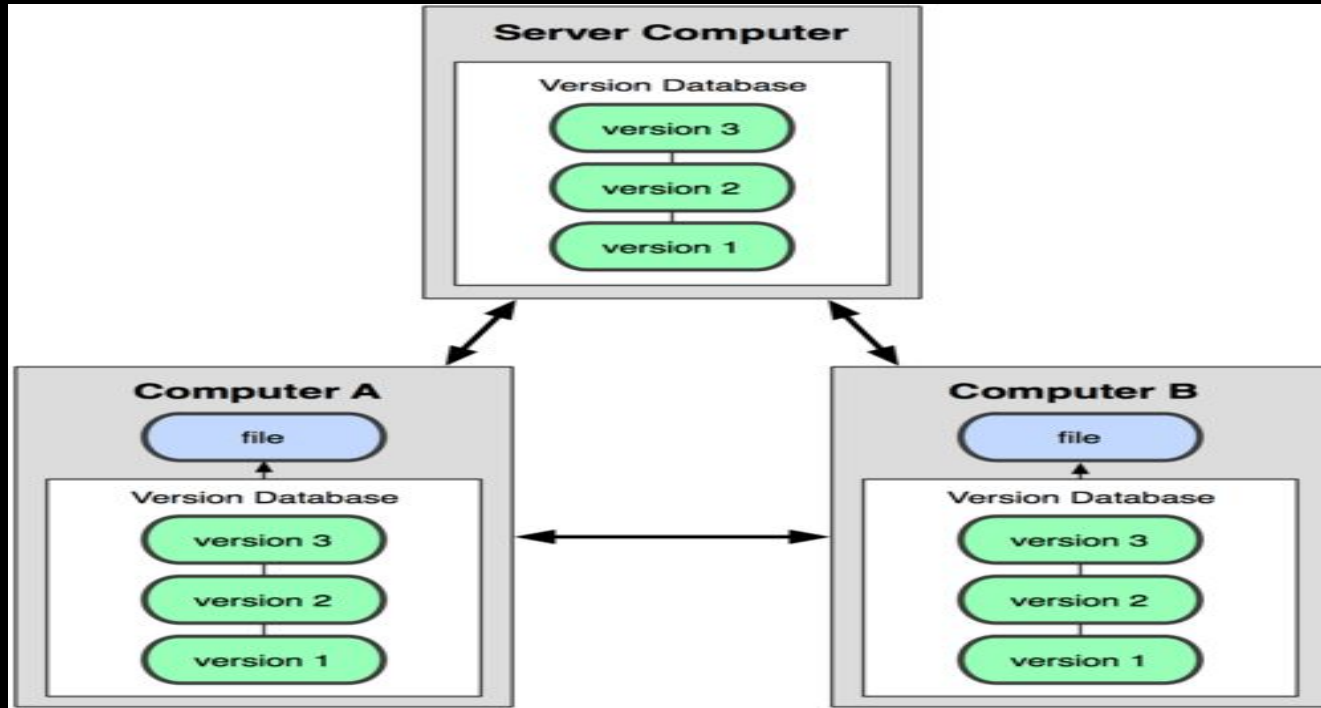
Cópias salvas localmente no computador.

2. CONTROLE DE VERSÃO CENTRALIZADO



Trabalho em conjunto.

3. CONTROLE DE VERSÃO DISTRIBUÍDO



Cópias completas do repositório salvas em um servidor.

FERRAMENTAS PARA TRABALHAR COM CONTROLE DE VERSÃO DISTRIBUÍDO:

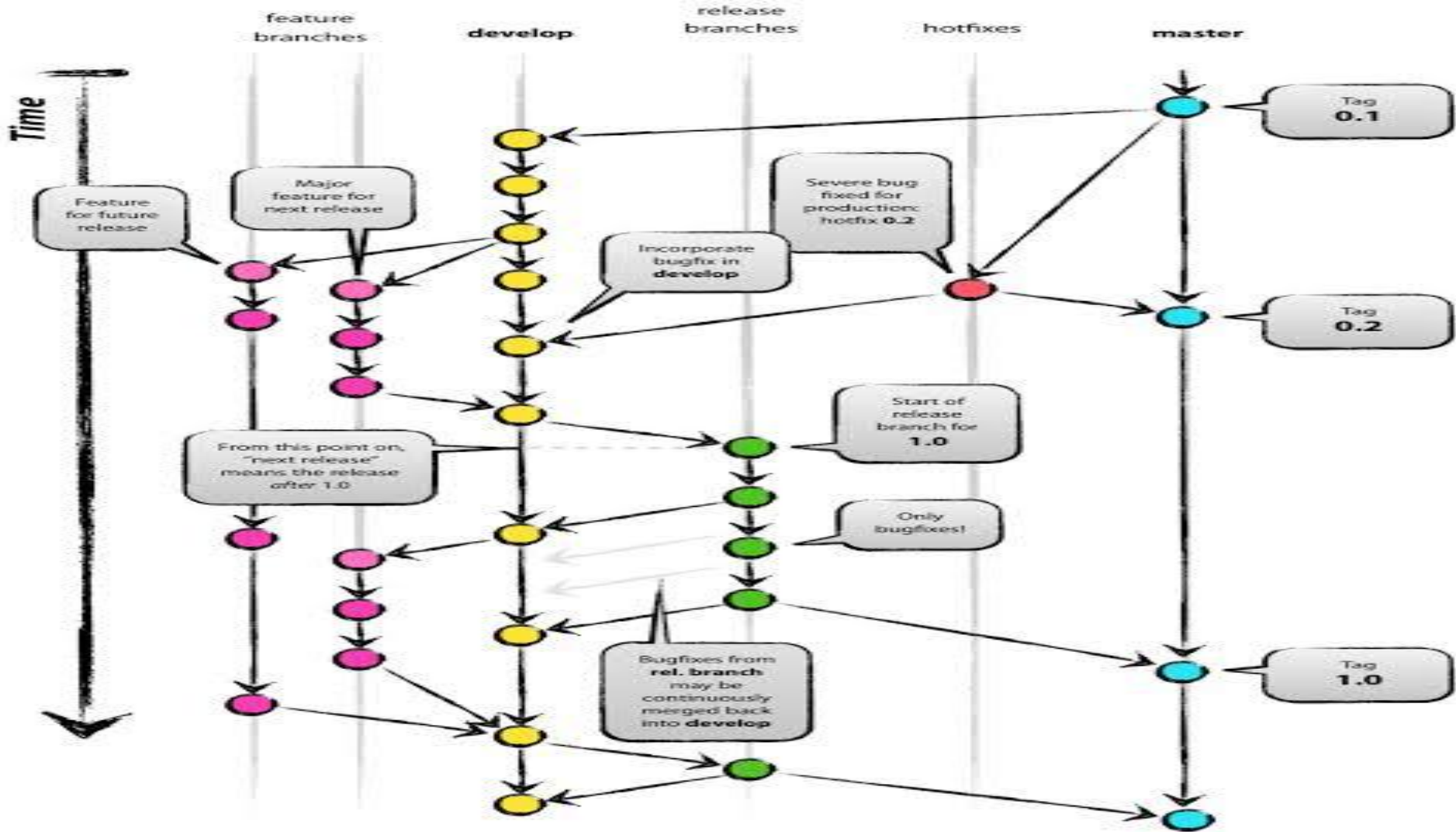


GIT

- Sistema de Controle de Versão Distribuído
- Desenvolvedor **Linux Torvalds** (*inicialmente desenvolvido para o Kernel Linux*)
- Milhares de **ramos paralelos**
- **Rápido e eficiente** com projetos grandes



FLUXO DE VERSIONAMENTO DE SOFTWARE (GIT FLOW)



PLATAFORMAS DE HOSPEDAGEM DE CÓDIGO-FONTE COM CONTROLE DE VERSÃO:



GitHub



GitLab



GITHUB

- Plataforma para **hospedagem de código fonte e colaboração com projetos privados e/ou Open Source**
- Mantido pela **Microsoft**
- **Fácil comunicação** em projetos
- Divulgar **principais trabalhos** (portfolio)



HORA DA PRÁTICA



REFERÊNCIAS

Documentação completa sobre Git:

- <https://git-scm.com/book/pt-br/v2>
- <https://pt.wikipedia.org/wiki/Git>

Sobre o GitHub:

- <https://pt.wikipedia.org/wiki/GitHub>
- <https://github.com/>

REFERÊNCIAS

Sobre fluxo de versionamento:

- <https://medium.com/trainingcenter/utilizando-o-fluxo-git-flow-e63d5e0d5e04>

Outros:

- <https://unsplash.com/>
- <https://www.iconfinder.com/>

OBRIGADA!