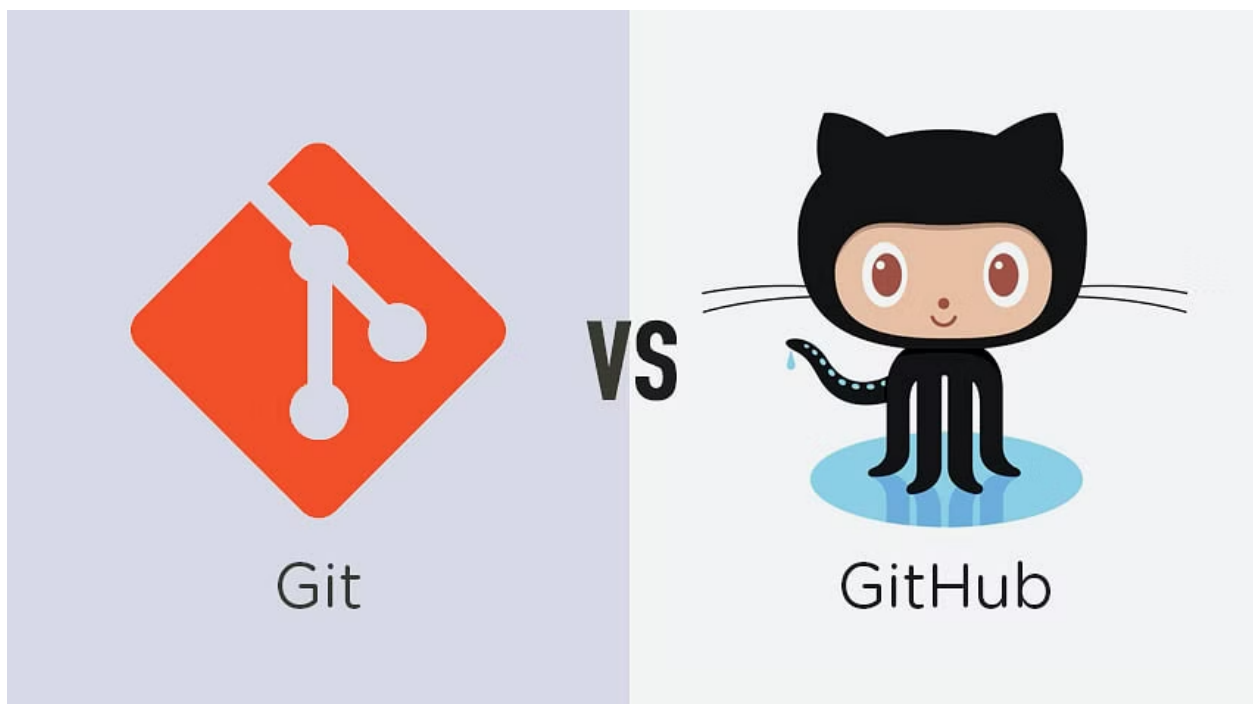


Git e GitHub

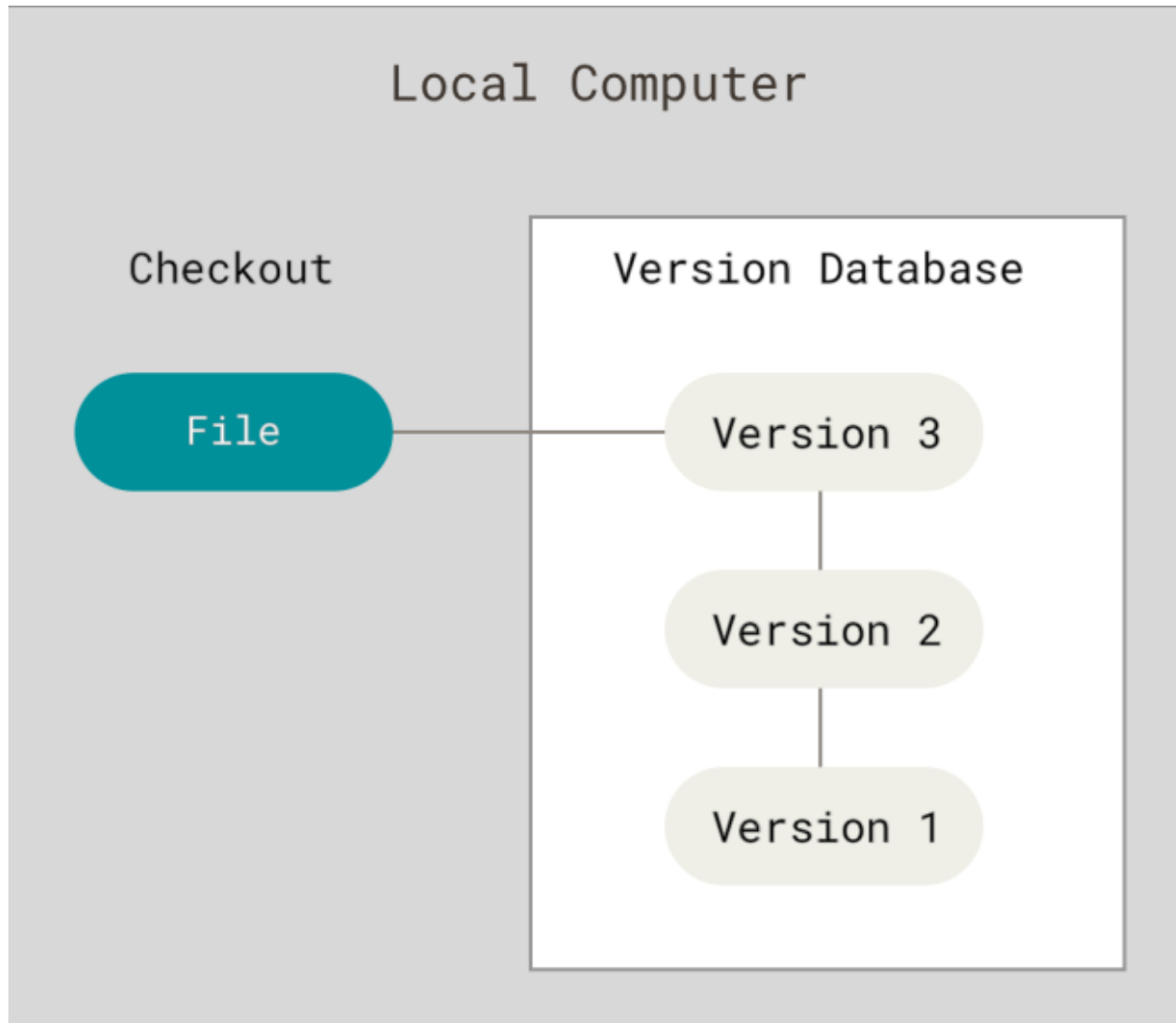


O que é o Git

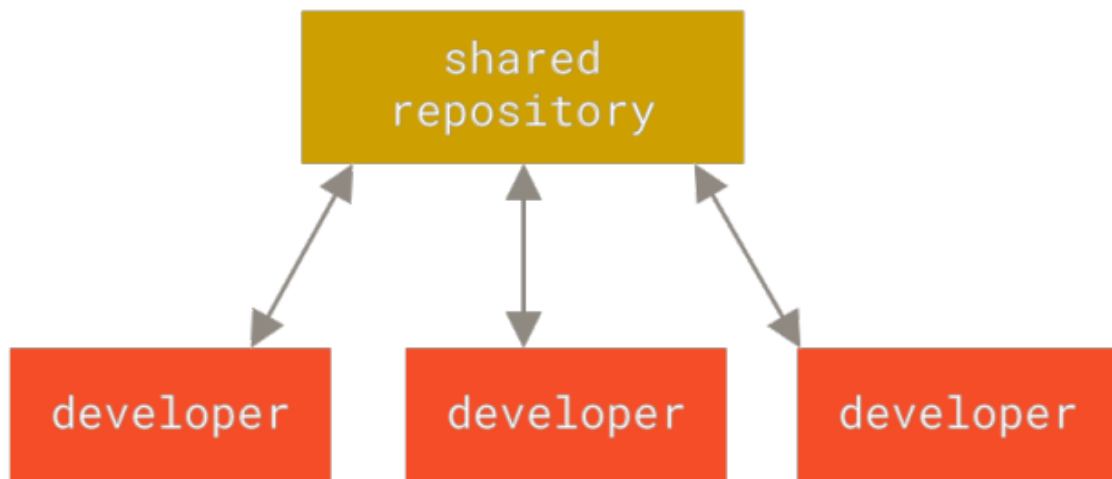
Um software criado pelo engenheiro de software Linus Torvalds, conhecido por ter desenvolvido, também, o núcleo Linux, o GIT é um Sistema de Controle de Versões Distribuído — ou DVCS(Distributed version control).

Vantagens de se usar o GIT

Possuir um controle de versão com a função de registrar quaisquer alterações feitas em cima de um código, armazenando essas informações e permitindo que, caso seja necessário, um(a) programador(a) possa regredir a versões anteriores de uma aplicação de modo simples e rápido.



Trabalhar de forma simultânea com outros desenvolvedores na mesma base de código, de forma que todos possam futuramente unir suas alterações mantendo o registro de qual desenvolvedor fez qual alteração.



Instalando

<https://git-scm.com/book/pt-br/v2/Começando-Instalando-o-Git>

<https://gist.github.com/oanhnn/80a89405ab9023894df7>

O que é o GitHub

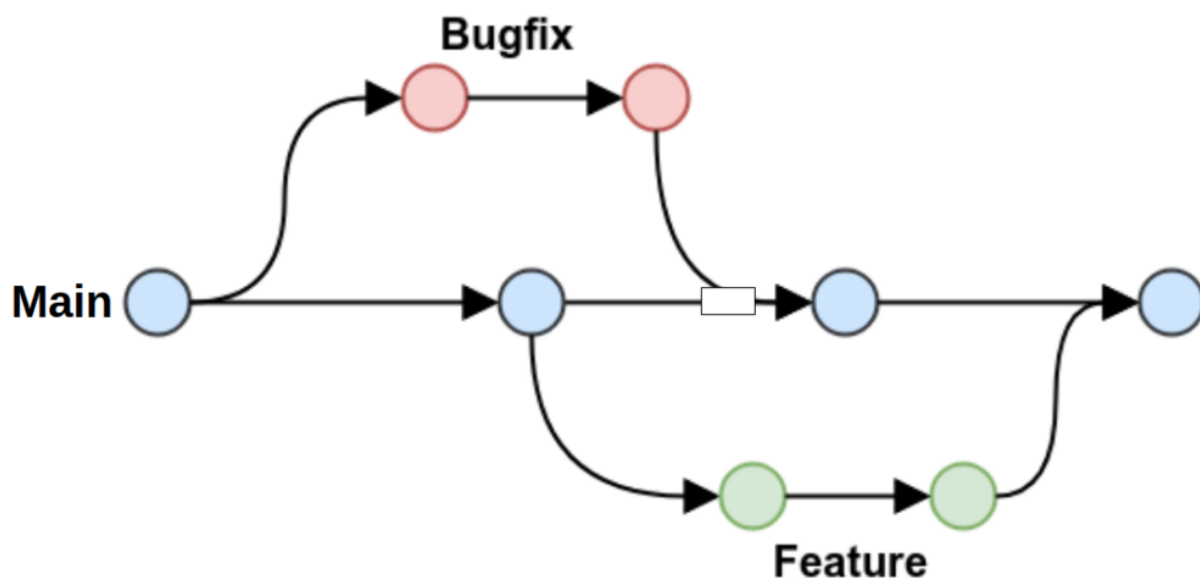
GitHub é uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo.

Prática

- Criar um repositório local
- Criar um arquivo chamado README.md
- Executar `$ git status`
- Adicionar esse arquivo `$ git add`
- Commitar esse arquivo `$ git commit -m "Primeiro commit"`
- Alterar o README.md
- Realizar o mesmo processo para commitar a mudança

- Criar uma conta no [GitHub](#)
- Criar um repositório
- Vincular o repositório
- Subir o repositório com `$ git push origin master`

Branchs



Quase todo Sistema de Controle de Versionamento tem alguma forma de suporte a ramificações (Branches). Ramificação significa que você diverge da linha principal de desenvolvimento e continua a trabalhar sem alterar essa linha principal. Em muitas ferramentas versionamento, este é um processo um tanto difícil, geralmente exigindo que você crie uma nova cópia do diretório do código-fonte, o que pode demorar muito em projetos maiores.

Prática

- Criar um nova branch `$ git branch -b add-new-txt`
- Criar um arquivo `lista_de_compras.txt` com 5 coisas que vc quer

- Executar `$ git status`
- Adicionar ele e realizar o commit
- Trocar entre a main e a nova branch com `$ git checkout main`
- Subir a nova branch para o GitHub `$ git push origin add-new-text`
- Criar um Pull Request add-new-txt → main
- Fazer o merge

Extra

Open Source

Código aberto (do inglês Open Source) é o código-fonte que é disponibilizado gratuitamente para consulta, examinação, modificação e redistribuição. Os produtos incluem permissão para usar o código-fonte, documentos de design ou conteúdo do produto. O modelo de código aberto é um modelo de desenvolvimento de software descentralizado que incentiva a colaboração aberta.

Projetos Open Source

<https://github.com/torvalds/linux>

He4rt Developers


A He4rt Developers é uma comunidade de desenvolvedores com o intuito de promover aulas, mentorias e debates sobre tecnologias em gerais! 🧑‍💻 Além disso, nossa comunidade possui projetos

🔗 <https://github.com/he4rt>



Google


You can't perform that action at this time. You signed in with another tab or window. You signed out in another tab or window. Reload to refresh your session. Reload to refresh your session.

 <https://github.com/google>



Globo.com

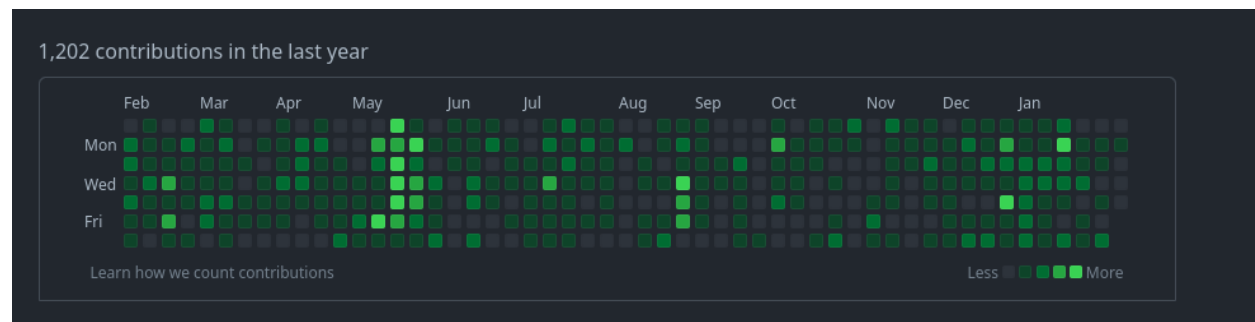
You can't perform that action at this time. You signed in with another tab or window. You signed out in another tab or window. Reload to refresh your session. Reload to refresh your session.

 <https://github.com/globocom>



Perfil

Contribuições



Exemplo de perfil com emblemas legais

