

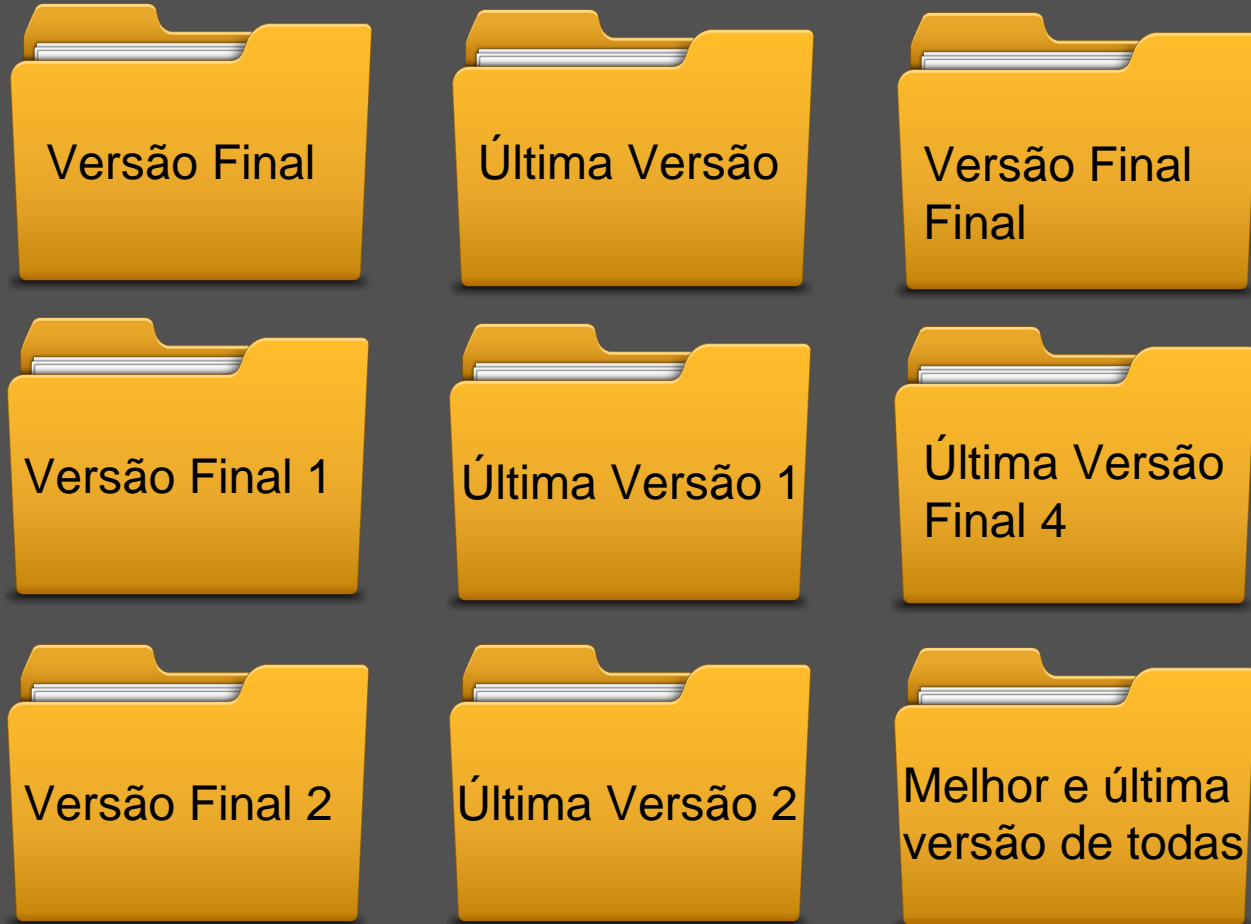


GIT

Prof. Me. Maicon dos Santos
Senac

Como você
organiza suas
mudanças?





GIT

- ❑ Desenvolvido/Criado em 2005 pelo Linus Torvalds (o homem conhecido por criar o núcleo, ou kernel, do SO Linux).
- ❑ GIT é um dos mais famosos sistemas de controle de versão de código aberto.

Companies & Projects Using Git

Google

facebook

Microsoft

twitter

LinkedIn

NETFLIX

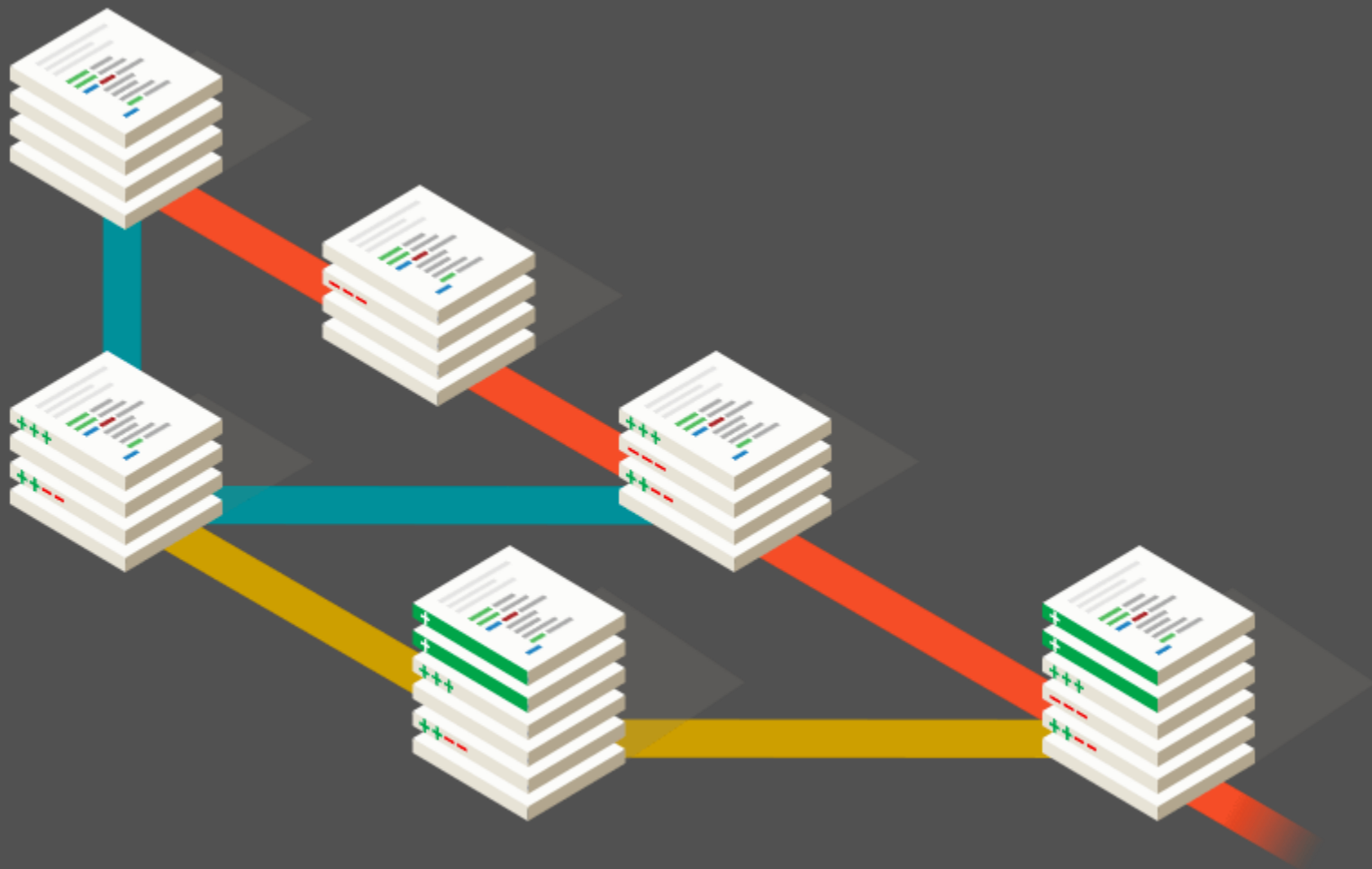


PostgreSQL



GIT

- ❑ Fluxo de trabalho básico Git:
 - ❑ Você modifica arquivos no seu diretório de trabalho.
 - ❑ Você adiciona arquivos modificados para uma área de transferência.
 - ❑ Você salva os arquivos que estão na área de preparo e armazena esses arquivos de forma permanente para o diretório Git.



Repositório
Interno



Um sistema de controle de
versão distribuído!

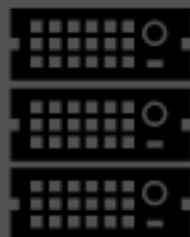
Repositório
Interno



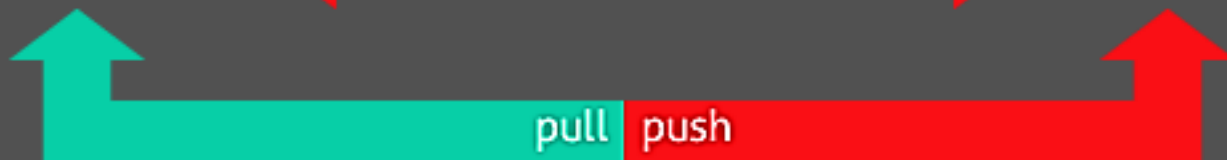
Workstation

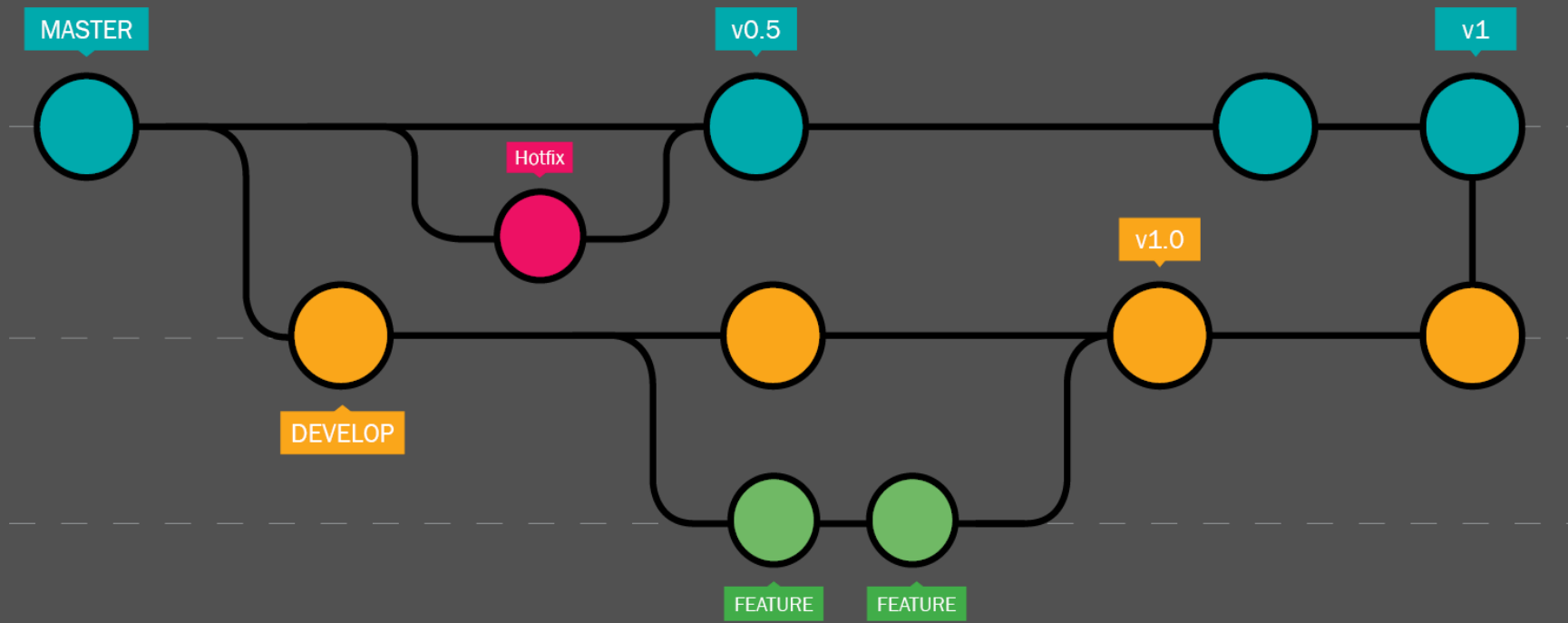


Servidor



Workstation





Instalação do GIT



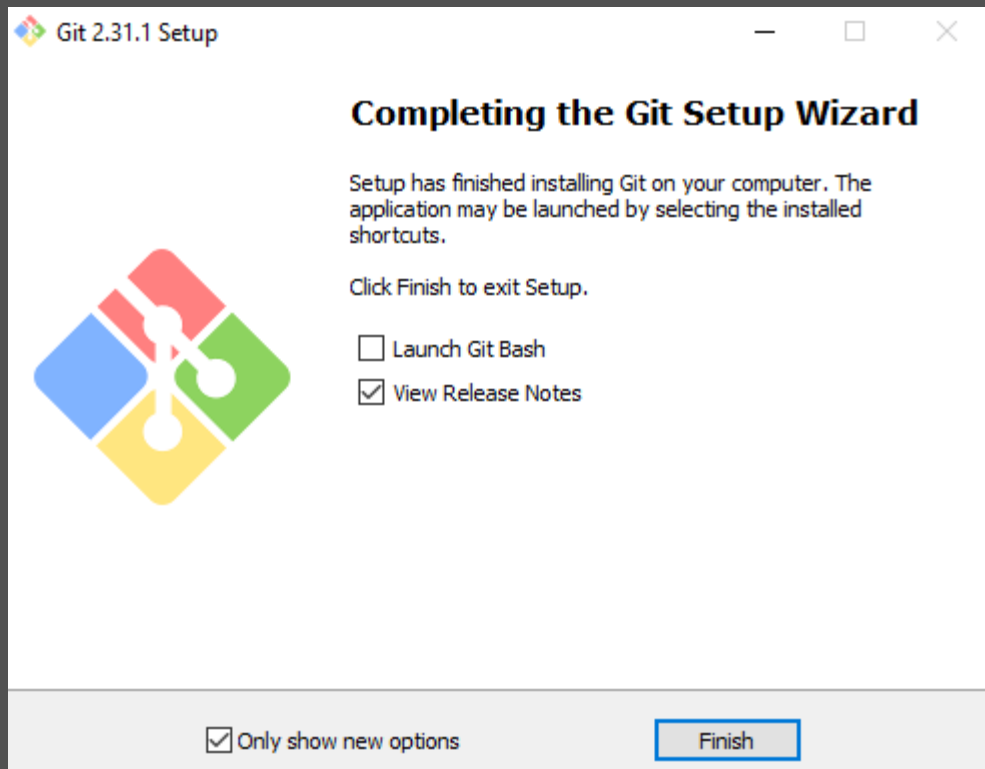
GIT

❑ Instalação do Git

- ❑ Para utilizarmos o Git, é fundamental torná-lo disponível em seu computador.
- ❑ <https://git-scm.com/download>
- ❑ Escolha o SO e mãos à obra!



Basta seguir as instruções na tela, clicando em **Next**.
Ao término, clique em **Finish** para concluir com êxito a instalação.



GIT

- Uma vez que o GIT está instalado e configurado no seu dispositivo, vamos explorar os conceitos básicos do GIT e como começar a usar o GIT.

GIT

Para configurar inicialmente seu repositório no GIT, é necessário executar o comando abaixo:

```
$ git init
```

Pronto, o projeto já é um repositório Git.

Uma pasta chamada .git foi criada no diretório em que o comando foi executado.

GIT

Porém, caso você queira obter a cópia de um repositório do Git já existente:

```
$ git clone url(repositório remoto)
```

Pronto, será obtido uma cópia/clone com todos os dados deste repositório remoto no diretório onde você executou o comando.

GIT

Para que os arquivos sejam versionados, você pode adiciona-los através do seguinte comando:

```
$ git add arquivo.txt
```

Será adicionado o arquivo “arquivo.txt” para preparar o commit e salvar o “arquivo.txt” no Git.

GIT

Podemos ver a situação dos arquivos no repositório Git acompanhando o status de cada modificação com o comando:

\$ git status

Será retornado os arquivos que foram adicionados anteriormente no comando git add e quando não adicionados informa os arquivos que não constam no repositório.

GIT

Para gravarmos as mudanças no repositório Git, devemos executar o comando:

```
$ git commit -m "Arquivo inicial"
```

Será salvo uma versão do “arquivo.txt” no repositório do Git. Será exibido mensagem que foi adicionado um *commit* identificado e ele carrega a seguinte mensagem de *commit* “Arquivo inicial”.

GIT

Para visualizar o histórico de commits existentes, basta utilizar o comando:

```
$ git log
```

Será listado todos os commits que foram realizados no repositório do Git. É um histórico!

GIT

Um recurso muito utilizado do GIT é sua capacidade de permitir que criem vários ramos (branches) independentes dentro de um único projeto.

Branches



GIT

O branch padrão em qualquer projeto é sempre o master branch. Um novo ramo (branch) pode ser criado usando o seguinte comando:

```
$ git checkout -b nome_branch
```

A *branch* será criada no **REPOSITÓRIO LOCAL**.

GIT

Porém caso você desejar retornar para uma *branch* existente, pode ser utilizado apenas o comando *checkout* da seguinte forma:

```
$ git checkout master
```

Seu repositório local agora será utilizado com os arquivos do repositório Git master.

GIT

Para tornar o *branch* disponível para outros usuários, você terá que **EMPURRAR** para o repositório remoto. Para fazer isso, use o seguinte comando:

```
$ git push origin nome_branch
```

A *branch* será enviada para o repositório remoto Git.

GIT

Caso você queira atualizar seu diretório de trabalho local para uma versão mais recente do repositório remoto, você pode **PUXAR** com o simples comando:

\$ git pull

\$ git fetch

Será atualizado o repositório local com os dados do repositório remoto do Git.

GIT

Agora que temos duas branches, para juntar duas branches, com diferentes *commits*, podemos unir as *branches* precisamos rodar o comando:

```
$ git merge nome_outra_branch
```

Para que o git merge funcione, precisamos estar na *branch* que irá receber os *commits*. Ao rodar o comando a branch será atualizada

GIT

Desfaz o commit do repositório remoto. Realiza a reversão de um commit de forma fácil e segura.
`git revert`

\$ git revert <commit>

O comando git revert sempre precisa de uma referência de commit para que seja revertido. Caso não seja informado, deverá passar o último commit como referência para seu correto funcionamento.

GIT

Desfaz as alterações nos arquivos de trabalho. Permite limpar por completo as alterações que não estão no repositório.

\$ git reset

Para que o git reset funcione é necessário que exista arquivos na branch adicionados.



Bit Bucket



GitHub



GitLab

NA PRÁTICA

GITHUB Desktop



GitHub, uma aplicação Web que possibilita criar de repositórios Git e de forma visual.



GitHub é um site no qual você pode fazer criação e atualização de projetos Git.
Crie uma conta no GitHub.

Acesse: <https://desktop.github.com/>



[Overview](#) [Release Notes](#) [Help](#)

GitHub Desktop

Focus on what matters instead of fighting with Git. Whether you're new to Git or a seasoned user, GitHub Desktop simplifies your development workflow.

Download for Windows (64bit)

Download for [macOS](#) or [Windows \(msi\)](#)

By downloading, you agree to the [Open Source Applications Terms](#).

IMPORTANTE!!!

A diferença entre GIT e GITHUB é que o GIT é uma ferramenta para versionar projetos, enquanto o GITHUB é a aplicação (site) no qual você colocará esses projetos Git versionados.

ATIVIDADES

1. Atividades com git:

- ✓ Crie uma conta no GitHub e um novo repositório;
- ✓ Faça o clone do repositório criado para o seu computador;
- ✓ Crie um arquivo README.md e faça o commit para o repositório;
- ✓ Faça o push das mudanças para o repositório no GitHub.

2. Continue a partir do exercício 1:

- ✓ Crie uma nova branch no seu repositório;
- ✓ Faça algumas mudanças em um arquivo existente no seu repositório;
- ✓ Faça o commit das mudanças na nova branch;
- ✓ Abra um pull request para mesclar a nova branch com a branch principal do repositório;
- ✓ Peça para um colega de equipe revisar o seu pull request;
- ✓ Se houver comentários ou alterações sugeridas, faça as mudanças necessárias e atualize o pull request;
- ✓ Depois que o pull request for aprovado, faça o merge da nova branch com a branch principal do repositório.

Aprendemos

- ✓ Instalar o Git,
- ✓ Criar repositórios,
- ✓ Adicionar arquivos,
- ✓ Verificar históricos,
- ✓ Visualizar estados,
- ✓ Fazer commits,
- ✓ Obter e Enviar versões,
- ✓ Criar branches,
- ✓ Navegar entre branches.



git

Obrigado!



Prof: Me. Maicon dos Santos