

## *Introdução ao git*

Olá, alunos(as),  
Bem-vindos(as) a nossa sexta aula da disciplina “Gerência de Configuração”. Nesta aula, estudar a ferramenta Git, conhecendo um pouco da sua história e da sua definição. Além disso, vamos ver como funciona o processo de instalação da nossa ferramenta, para que possamos fazer os nossos primeiros trabalhos.

Assim, não se esqueçam de ler esta aula, assistir os nossos vídeos e fazer as atividades. Se tiverem alguma dúvida, vocês podem usar o nosso quadro de avisos, que eu responderei.

⚡ Bons estudos!

### **Objetivos de** aprendizagem

Ao término desta aula, vocês serão capazes de:

- Conhecer o que é o Git;
- Saber a história da ferramenta;
- Entender como instalar o Git.

## Seções de estudo

- 1 – O que é o git
- 2 – Como instalar o git

### 1 - O que é o git

Vimos, na aula 3, como funciona o Gerenciamento de Versões enquanto atividade da Gerência de Configuração. Mas ainda não vimos de fato um gerenciador de versões. A partir desta aula, vamos estudar o Gerenciador de Versões Git, que é um dos gerenciadores de versão mais populares do mundo.

O Git é um sistema de controle de versão que, pela sua estrutura interna, é uma máquina do tempo extremamente rápida e é um robô de integração bem competente.

*Fonte: AQUILES; FERREIRA, 2017, p. 3*

Vamos ver um pouco de história. Em 2005, Linus Torvalds, criador do kernel<sup>1</sup> do Linux, enfrentou um dilema. O código-fonte do kernel do Linux estava hospedado no sistema de Gerenciamento de Versões proprietário denominado de BitKeeper. Mas, houve desentendimentos e insatisfações entre a equipe que mantinha o kernel e a equipe do BitKeeper.

1 - Núcleo do sistema operacional.

Figura 1 - Linus Torvalds, criador do Linux



*Fonte: Disponível em: [https://upload.wikimedia.org/wikipedia/commons/0/01/LinuxCon\\_Europe\\_Linus\\_Torvalds\\_03\\_%28cropped%29.jpg](https://upload.wikimedia.org/wikipedia/commons/0/01/LinuxCon_Europe_Linus_Torvalds_03_%28cropped%29.jpg). Acesso em 24 set. 2019.  
Autores: Krd & Von Sprat.*

A gota d'água da relação foi uma mudança na estratégia de monetização da ferramenta. Até então, a equipe do Linux não era obrigado a pagar pelo uso da ferramenta devido à natureza do sistema. Porém, a isenção de uso da ferramenta foi revogado, e a equipe mantenedora teria que desembolsar dólares para manter o código-fonte hospedado nessa ferramenta.

Assim, Linus Torvalds tomou uma decisão nada fácil: decidiu que iria começar um novo projeto: Criar um gerenciador de versões mais simples e rápido do que os existentes no mercado (Naquela época, os gerenciadores de versão CVS e Subversion lideravam o mercado).

Para projetar esse novo sistema, foram levadas em conta as lições que a equipe aprendeu a operar o Bitkeeper. Assim, os objetivos que pretendiam atingir com o Git eram:

- Velocidade
- Design simples
- Suporte robusto a desenvolvimento não linear (milhares de branches paralelos)
- Totalmente distribuído
- Capaz de lidar eficientemente com grandes projetos como o kernel do Linux (velocidade e volume de dados)

Esse último item é importante. No ano de 2018, o projeto do kernel do Linux teve mais de 75 mil commits (vamos explicar o que é mais tarde). Ao todo esses commits adicionaram 3.385.121 linhas de código e removeram 2.512.040 linhas de código.

A versão 4.13 do kernel tem a impressionante marca de 20 milhões de linhas de código.

Em 21 de Dezembro de 2005 (LWN, 2005), foi lançado a versão 1.0.0 do sistema, que ganhou uma rápida adoção entre os entusiastas do software livre. Muito se deve ao fato de que o nome de Linus Torvalds e o seu uso no projeto do kernel do Linux acabou ajudando na propagação do sistema.

Além disso, outro fator que foi decisivo pela popularização do Git foi a criação de plataformas que hospedam repositórios git e fornecem outros serviços. O mais famoso desses serviços é o Github. Fundado em 2008, a plataforma atingiu a marca de 100 milhões de repositórios sob os seus cuidados no ano de 2018 (WARNER, 2018).

Como você vai ver mais adiante, Git e Github são coisas diferentes.

Atualmente, o Git se encontra na versão 2.23.0 (16/08/2019), com uma enorme adoção e popularidade.

#### Que fim levou o BitKeeper?

Bem, o BitKeeper agora é um Gerenciador de Controle de Versões Open Source. Mais informações sobre ele podem ser encontradas aqui: <http://www.bitkeeper.org/>

**BitKeeper** is the original distributed source management system.  
 Now available as Open Source under the Apache 2.0 License.

BitKeeper is a fast, enterprise-ready, distributed SCM that scales up to very large projects and down to tiny ones.

Download Now!  
 BitKeeper 7.3.3  
 (Binaries and Source)

#### Features

- **Simple:** An easy to use command line interface.
- **Scalable:** Nested Repositories are submodules done right! Version control collections of repositories.
- **Flexible:** Hybrid mode for binary files that uses a cloud of server for binaries instead of bloating the source repositories.
- **Accurate:** Tracking of file operations like creates, deletes, and renames.
- **Safe:** All file accesses validate checksums for integrity. All file writes include redundancy for error correction.
- **Dependable:** Highly accurate auto-merge that uses the whole history to resolve conflicts. Most other systems use variations of `diff3`.
- **Discernable:** Source annotations instantly available.
- **Fast:** High performance and scales to very large repositories.
- **Free:** Licensed under the [Apache Version 2 license](#)

#### Clone a project and push changes

```
$ bk clone bk://bkbkbits.net/bkdemo/bk_demo
$ cd bkdemo
# edit files using your favorite editor
$ bk -Ux new
$ bk commit -y "Comments"
$ bk push
```

#### Create a project and commit

```
$ bk init new_project
$ cd new_project
# edit files using your favorite editor
```

Agora que você sabe um pouco da história da ferramenta, vamos ver como funciona o processo de instalação.

## 2 - Como instalar o git

Vamos agora mostrar como instalar o Git no seu computador. A instalação do Git é fundamental, pois sem ele, não conseguiremos trabalhar em nossos repositórios.

Se você estiver usando o Linux, pode ser que a distribuição já tenha embutido o Git por padrão. Para verificar, digite em seu terminal o comando `git --version`. Se receber alguma resposta do Git, nada precisa ser feito. Caso contrário, procure pelo pacote git em seu gerenciador de dependências da sua distribuição.

Se você estiver usando uma distribuição baseada no Debian (Debian, Ubuntu - e seus sabores, Linux Mint, etc), digite os seguintes comandos em uma janela do terminal - você será solicitado a digitar a senha da sua conta:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install git
```

Caso você esteja usando uma distribuição baseada no Red Hat (Red Hat, SuSE, Fedora, CentOS...) digite os seguintes comandos em uma janela de terminal - você será solicitado a digitar a senha da sua conta:

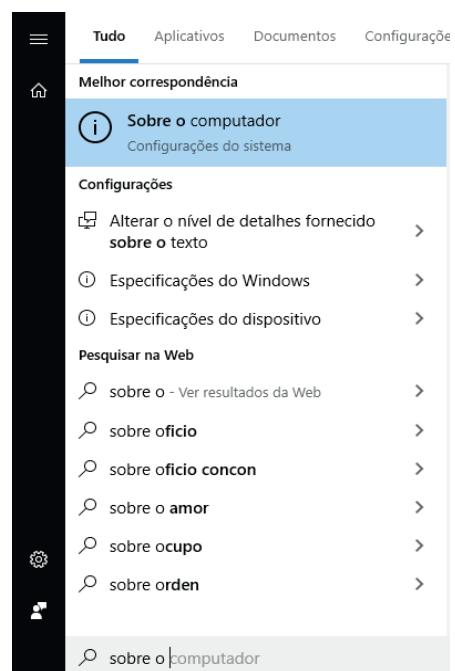
```
sudo yum upgrade
sudo yum install git
```

Caso você esteja usando o MacOS, você deve instalar o gerenciador de pacotes Homebrew ([https://brew.sh/index\\_pt-br](https://brew.sh/index_pt-br)). Após instalar, você deve digitar o seguinte comando no terminal e dar Enter:

```
brew install git
```

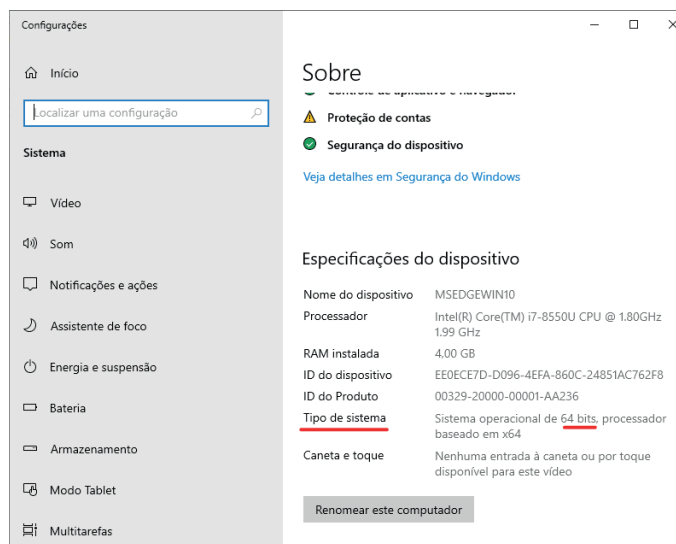
Caso o seu sistema seja Windows, você deve baixar o instalador e executar -lo. Nesta seção, vamos abordar os passos de instalação. A nossa máquina, na qual o Git será instalado, é uma máquina virtual com Windows 10 1809 instalado.

Primeiramente, vamos ver se o nosso sistema é de 32 ou de 64 bits. Se você estiver no Windows 10, pesquise no menu iniciar "Sobre o Computador" ou "About this Computer". Clique na primeira opção que aparecer:



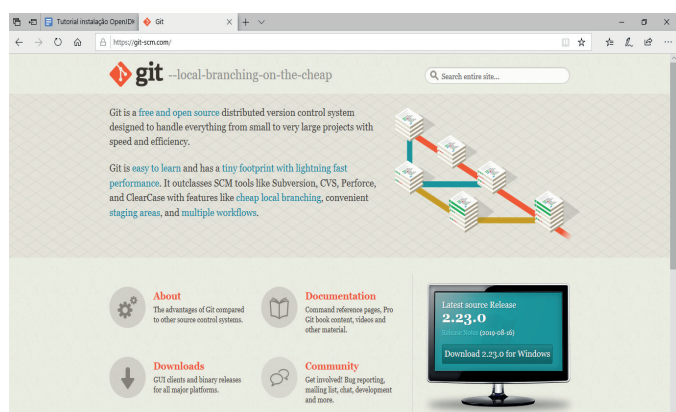
Na janela que surgir, procure o item Especificações do Dispositivo. Nesta seção, procure a linha Tipo de Sistema.

Nele você vai encontrar o tipo do sistema:



Feito isso, podemos instalar o Git.

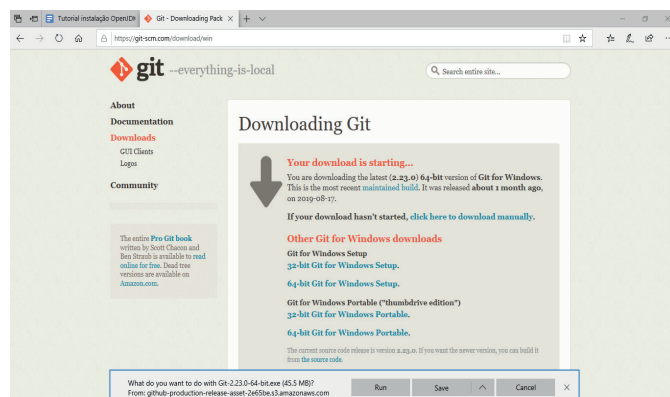
Passo 1: Acesse o site oficial do Git: <https://git-scm.com/>



No site há um monitor informando a versão mais recente do Github. Nesse monitor, há um botão escrito “Download X.XX.X for Windows”. Clique nele.



Nesse caso, o Git irá detectar a versão do seu sistema operacional e irá iniciar o download automaticamente. Salve e baixe o arquivo, seguindo as instruções do seu navegador. Aguarde a conclusão do download.



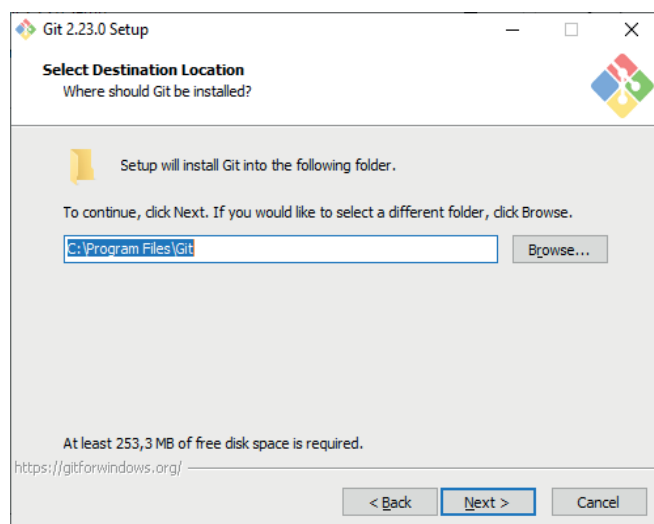
Passo 2: Ao terminar de fazer o Download, execute o instalador.

Quando abrir o instalador, pode ser que o Windows peça para que você confirme a execução. Confirme-a.

A primeira tela é a apresentação da licença do Git. Essa licença é a licença pública GNU, que permite que você use o Git gratuitamente tanto no contexto pessoal, quanto no contexto profissional. Leia a licença e confirme, clicando em Next.



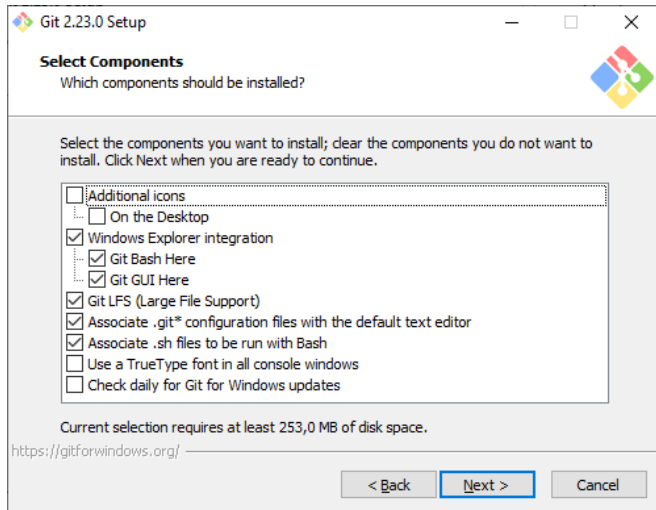
Em seguida, o Git vai perguntar onde vai instalar o programa. Deixe como está e clique em “Next >”:



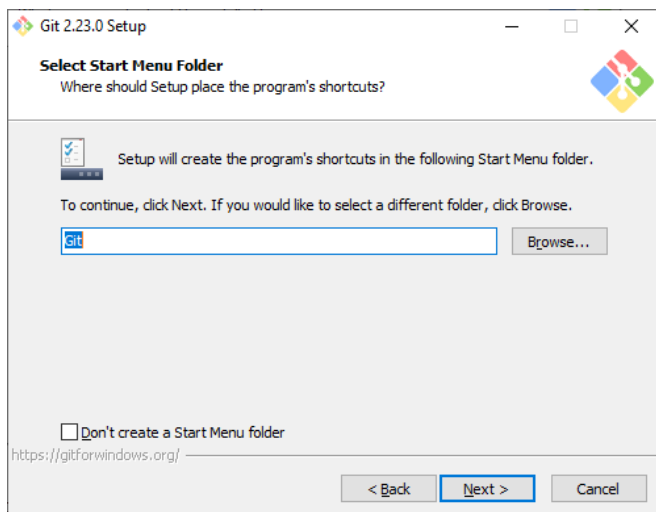
A próxima tela indaga quais componentes você quer



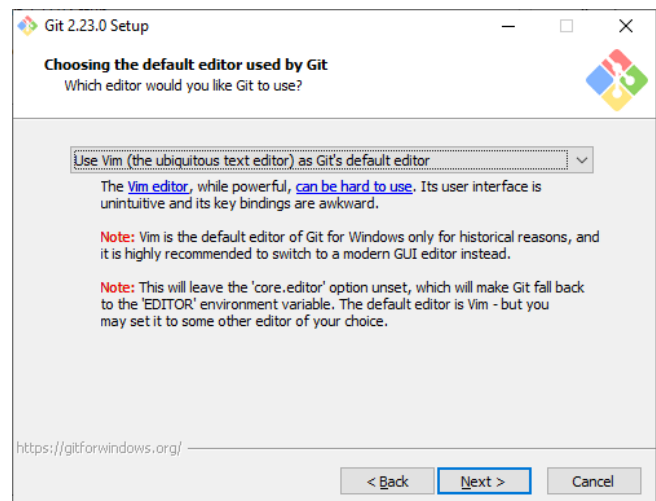
instalar no Git. Deixe também como está e clique em “Next >”:



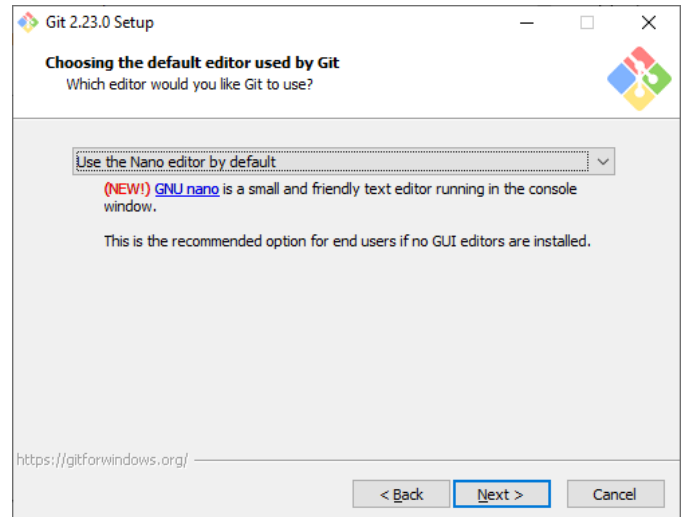
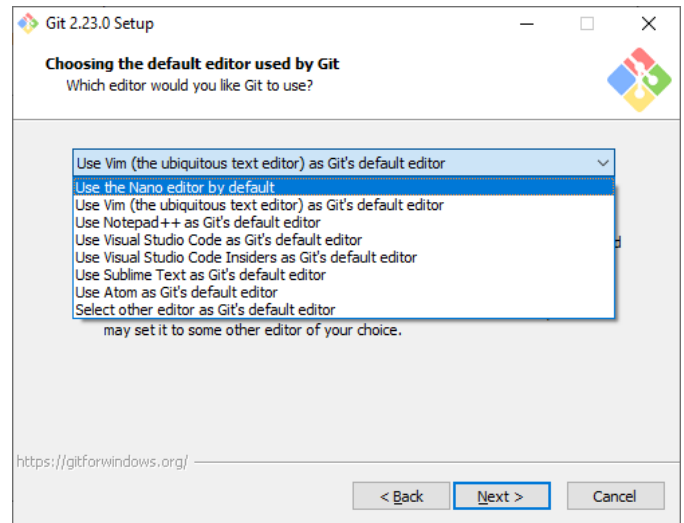
O próximo item indica qual é a pasta do Menu Iniciar que o programa de instalação vai colocar os atalhos para os programas do Git. Novamente, deixe como está, e clique em “Next >”:



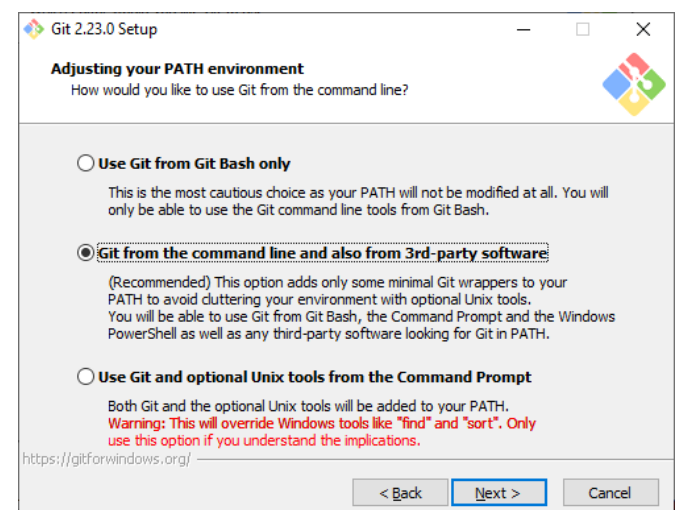
O próximo item é importante. Ele define qual vai ser o editor de arquivos padrão do Git. Esse editor será usado para escrever as mensagens dos nossos commits. Por padrão, o Git indica o Vim, um editor que é comum de se encontrar no Linux. Mas, é um pouco complicado de se usar.



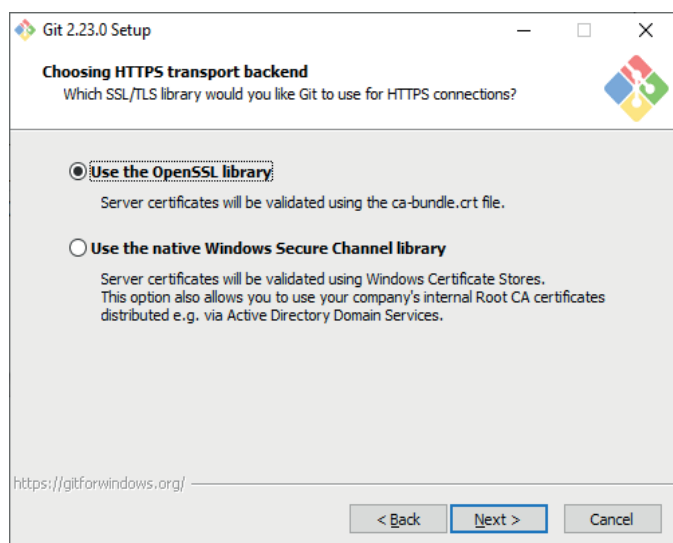
Assim, para facilitar a sua vida, podemos trocar o editor. Vamos trocar pelo Nano, um editor de texto por linha de comando, mais simples do que o Vim, que será instalado junto com o Git (assim como o Vim). Para isso, clique na combobox e selecione a opção “Use the Nano editor by default”. Depois, clique em “Next >”:



Em seguida, o instalador vai perguntar quais aplicativos serão adicionados ao PATH do sistema. Por padrão, o Git indica como recomendável a opção de colocar apenas algumas ferramentas do Git (incluindo o próprio Git) no PATH. Assim, não mexa em nada e clique em “Next >”:



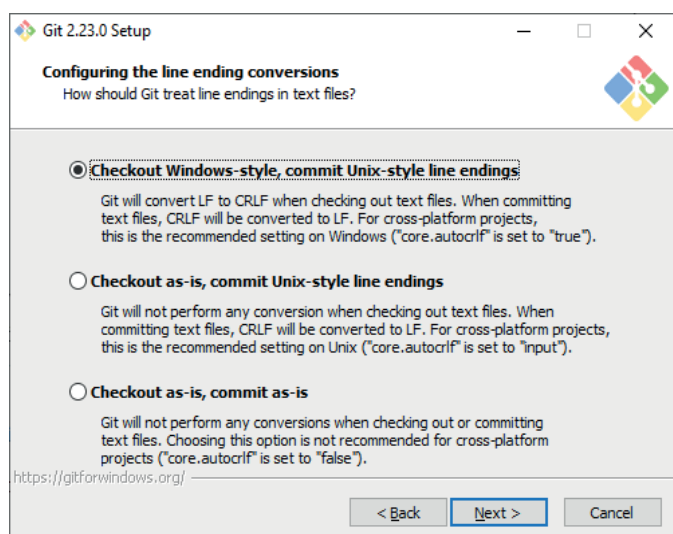
Em seguida, o instalador irá perguntar de que forma serão feitas as conexões seguras com os repositórios do Git. Deixe também do jeito que está – por questões de compatibilidade – e clique em “Next >”.



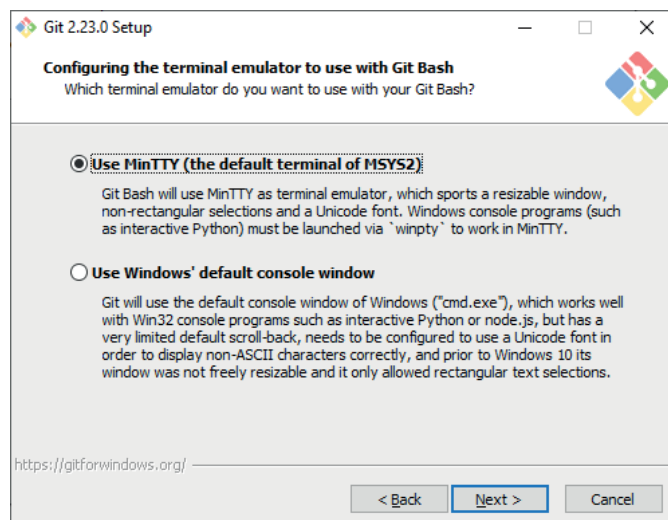
A próxima tela do instalador pergunta de que forma o Git irá tratar as quebras de linha. Como você deve saber, os sistemas Unix têm uma forma diferente de tratar as quebras de linha em relação ao Windows.

A opção padrão do instalador define que quando você recuperar uma versão de um arquivo de texto (checkout), o Git irá converter as quebras de linha para o padrão do Windows. E quando você salvar as alterações no repositório – commit, o Git irá converter as quebras de linha para o padrão do Unix.

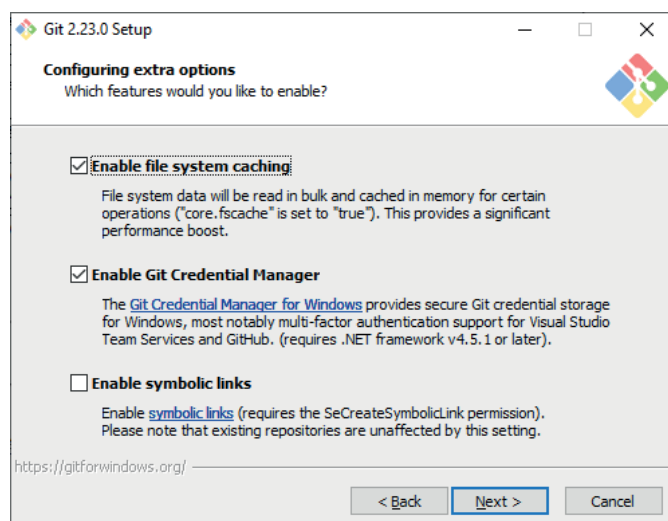
Segundo a equipe de desenvolvimento, essa é a melhor opção. Assim, vamos deixar como está e clicar em “Next >”:



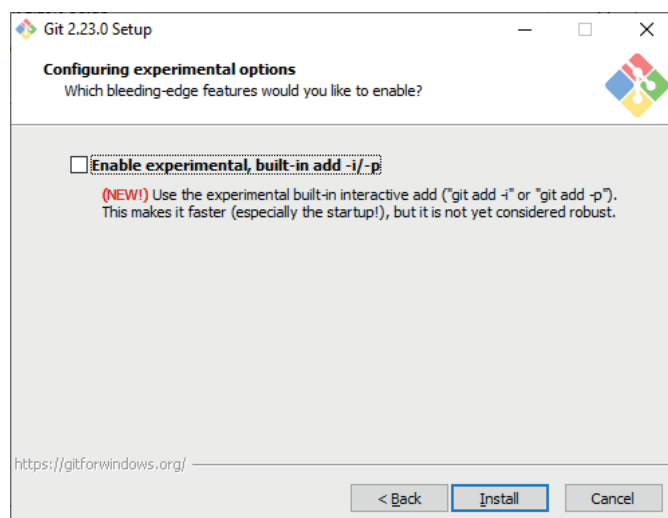
A próxima tela pergunta qual terminal você deseja usar para o Git Bash. Se você estiver usando uma versão anterior ao Windows 10, você deve marcar a primeira opção. Caso esteja usando o Windows 10, selecione uma das duas opções. Clique em “Next >”.



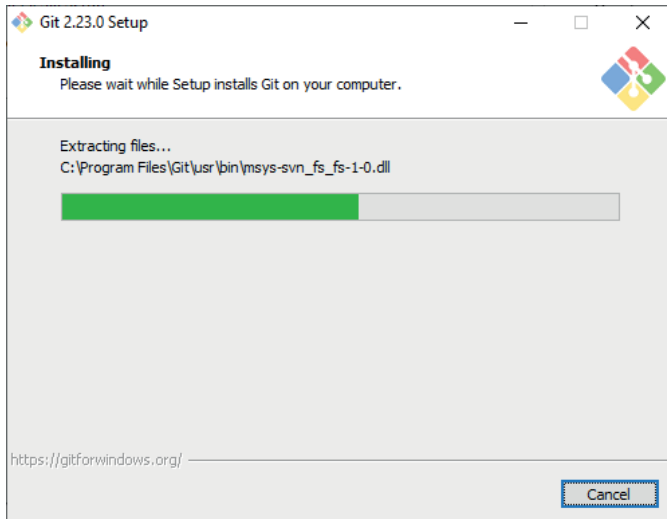
Na próxima tela, o Git oferecerá uma série de configurações. Deixe como estão as opções (ou seja, não mexa em nada) e clique em “Next >”:



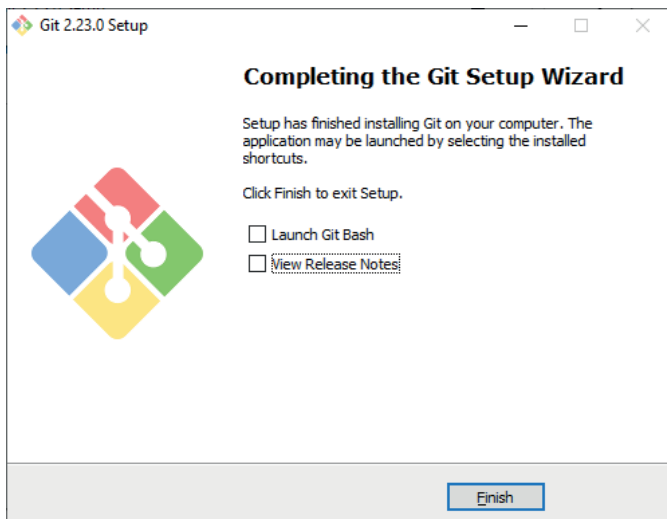
Em seguida, o Git perguntará se deseja ativar uma funcionalidade avançada experimental. Deixe desmarcado. Clique em “Next >”:



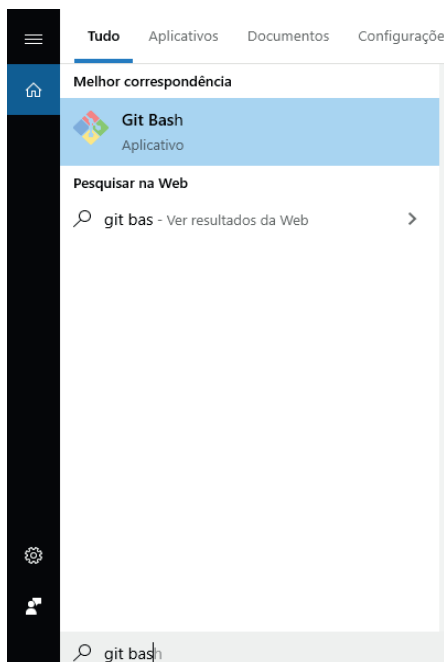
Aguarde a conclusão da instalação.



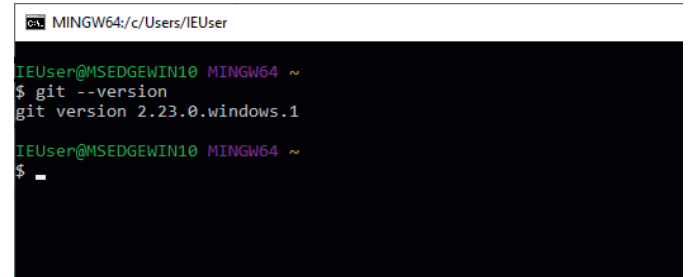
Ao final da instalação, desmarque as opções e clique em “Finish”:



Agora, vamos ver se o Git foi corretamente instalado. Procure no menu iniciar por Git Bash:



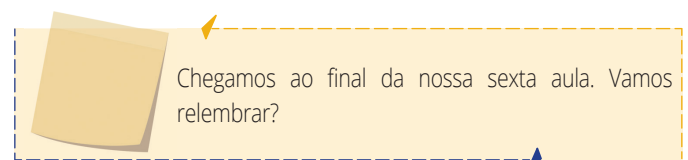
Vai aparecer um console. Digite o comando `git --version` para confirmar a instalação do Git:



Se você vir uma mensagem como “git version x.xx.x”, parabéns! O Git está instalado!

E com isso, encerramos a nossa sexta aula. Na próxima aula, vamos abordar como funciona o processo de criação de repositório e população dos commits. Até lá!

## Retomando a aula



### 1 – O que é o git

Nesta seção, você viu que o Git é uma ferramenta desenhada para o controle de versões de maneira eficiente. Foi concebido por Linus Torvalds e uma comunidade de programadores voluntários, tendo em mente os desafios de gerenciar o Kernel do Linux.

### 2 – Como instalar o git

Nesta seção, você viu as instruções de instalação do Git para máquinas Windows. Como podemos ver, o roteiro de instalação não é tão diferente quando vemos para outros programas Windows, mas demanda atenção em alguns itens.

## Vale a pena

‘Vale a pena ler,



AQUILES, Alexandre; FERREIRA, Rodrigo. *Controlando versões com Git e GitHub*. São Paulo: Casa do Código, 2017.



### Vale a pena **acessar**,

DIOLINUX. Linux 5.2 chega com quase 600 mil novas linhas de código. 2019. Disponível em: <https://www.diolinux.com.br/2019/07/kernel-52-chega-com-quase-600-mil-novas-linhas-de-codigo.html>. Acesso em 19 nov. 2019.

GIT. Uma Breve História do Git. s.d. Disponível em: <https://git-scm.com/book/pt-br/v1/Primeiros-passos-Uma-Breve-Hist%C3%B3ria-do-Git>. Acesso em 19 nov. 2019.

GROW UNDER. Git: manual de utilização e vantagens e desvantagens. 2016. Disponível em: <https://www.growunder.com/pt/blog/dicas/65-git-como-funciona-e-quais-as-vantagens-e-desvantagens>. Acesso em 19 nov. 2019.

LWN. Git 1.0.0 released. 2005. Disponível em: <https://lwn.net/Articles/165127/>. Acesso em 19 nov. 2019.

WARNER. Thank you for 100 million repositories. Github, 2018. Disponível em: <https://github.blog/2018-11-08-100m-repos/>. Acesso em 19 nov. 2019.



### Minhas anotações