


Aula #2: Instalação do Git (Windows)

Download do Git	1
Processo de Instalação	3
Configurações Iniciais	13
Próxima Aula	14
Fontes e Links Complementares	14


Download do Git


O processo de download do versionador de códigos Git é muito simples: basta acessar o [site](#) da ferramenta e clicar no link de Download.

 **git** --fast-version-control

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.


Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.






About

The advantages of Git compared to other source control systems.




Documentation

Command reference pages, Pro Git book content, videos and other material.



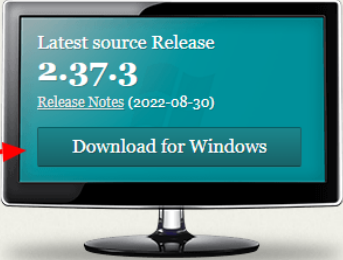
Downloads

GUI clients and binary releases for all major platforms.




Community


Get involved! Bug reporting, mailing list, chat, development and more.





Latest source Release
2.37.3
[Release Notes \(2022-08-30\)](#)
[Download for Windows](#)




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

 [Windows GUIs](#)

 [Tarballs](#)

 [Mac Build](#)

 [Source Code](#)

Na próxima tela, basta selecionar a versão 64 bits, caso o seu sistema seja 64 bits, ou 32, caso não seja:

git --fast-version-control

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](#).

Download for Windows

[Click here to download](#) the latest (2.37.3) 64-bit version of **Git for Windows**. This is the most recent [maintained build](#). It was released **13 days ago**, on 2022-08-30.

Other Git for Windows downloads

Standalone Installer

[32-bit Git for Windows Setup.](#)

[64-bit Git for Windows Setup.](#) ←

Portable ("thumbdrive edition")

[32-bit Git for Windows Portable.](#)

[64-bit Git for Windows Portable.](#)

Using winget tool

Install [winget tool](#) if you don't already have it, then type this command in command prompt or Powershell.

```
winget install --id Git.Git -e --source winget
```

The current source code release is version **2.37.3**. If you want the newer version, you can build it from [the source code](#).

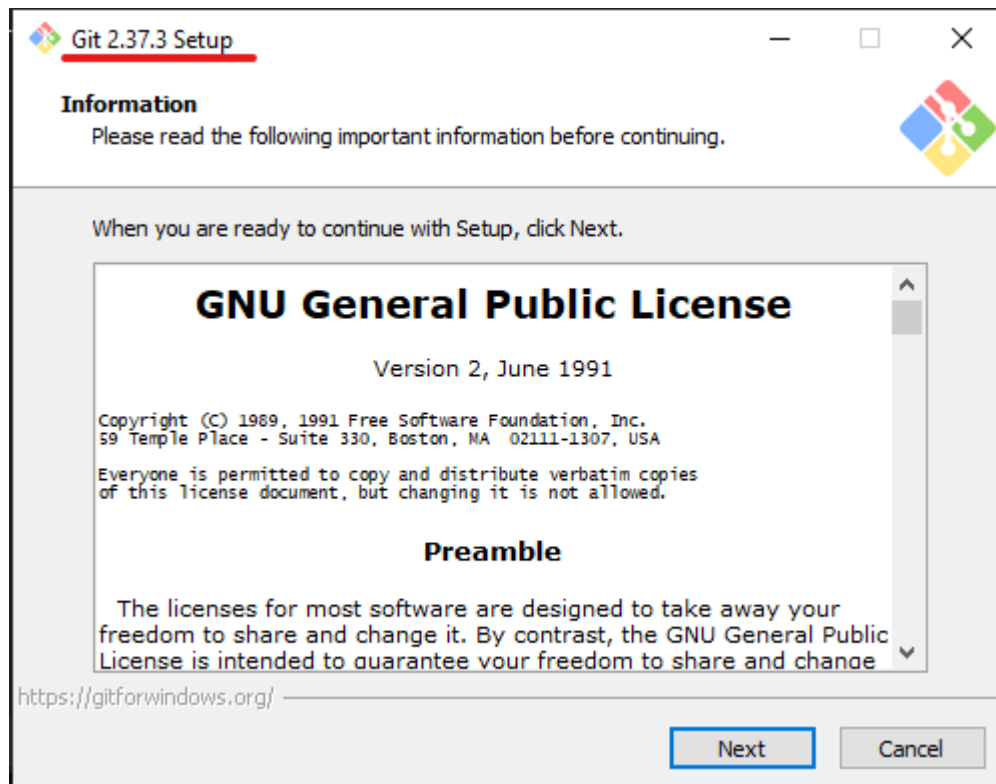
Now What?

Now that you have downloaded Git, it's time to start using it.

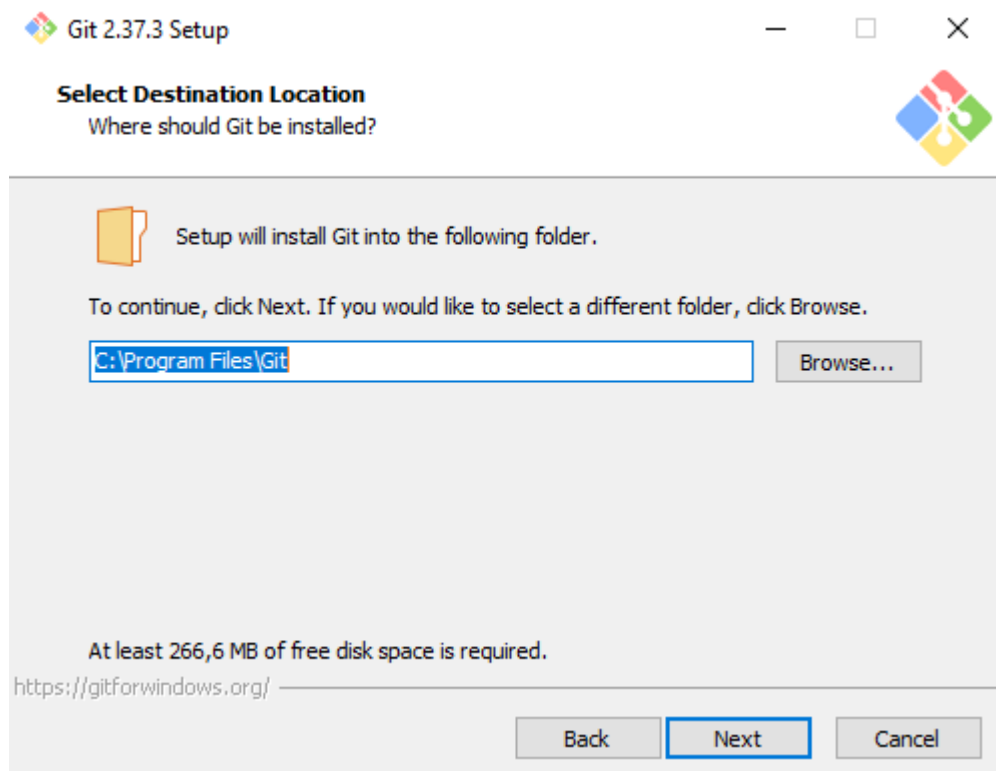
O Git em si é leve e deve ser baixado rapidamente. Obviamente, tudo vai depender da velocidade da sua conexão.

Processo de Instalação

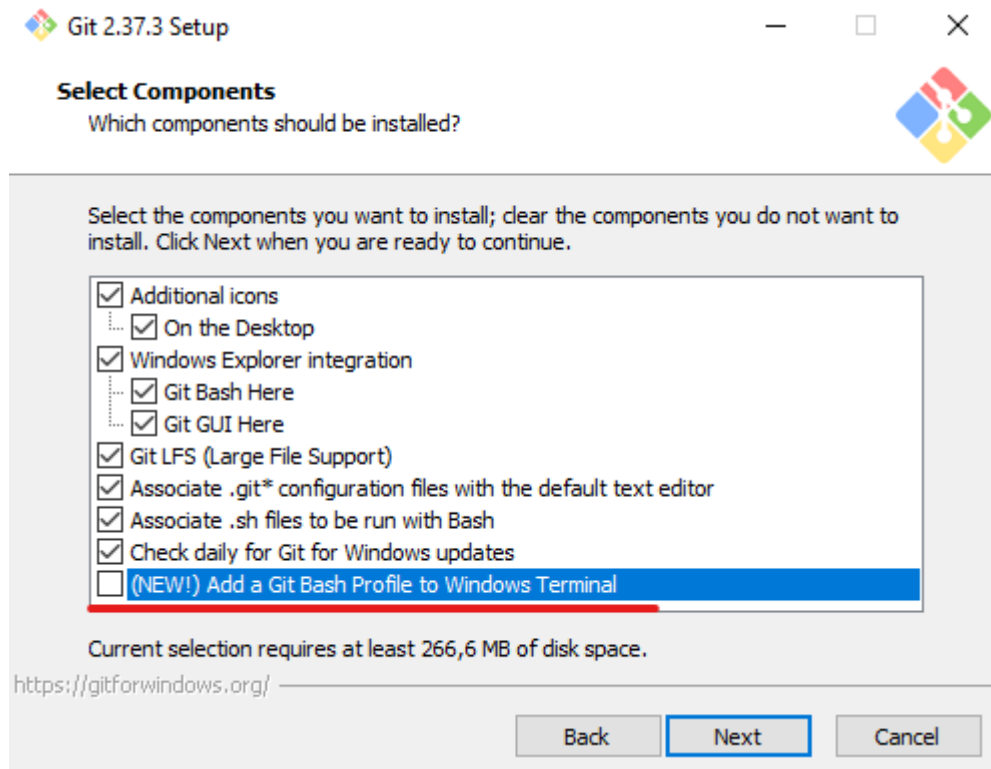
Assim como o download, o processo de instalação é simples e rápido. Primeiro execute o instalador baixado para abrir o instalador. Uma informação importante antes de iniciarmos o processo, é que a versão utilizada na criação dessa aula é a versão 2.37.3, e as opções de instalação podem ser modificadas com a atualização do instalador.



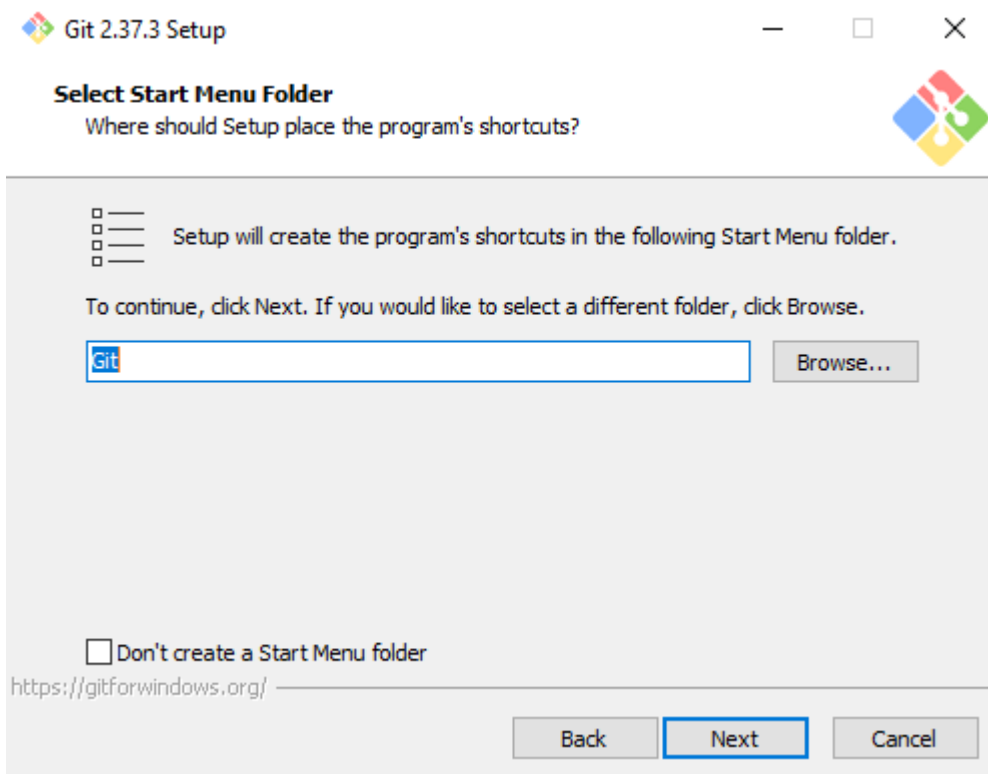
Uma vez com o instalador aberto, basta clicar em "Next". Na próxima tela, será pedido para selecionarmos o caminho da instalação. Vamos deixar o caminho padrão, ou seja, o caminho pré-selecionado pelo instalador.



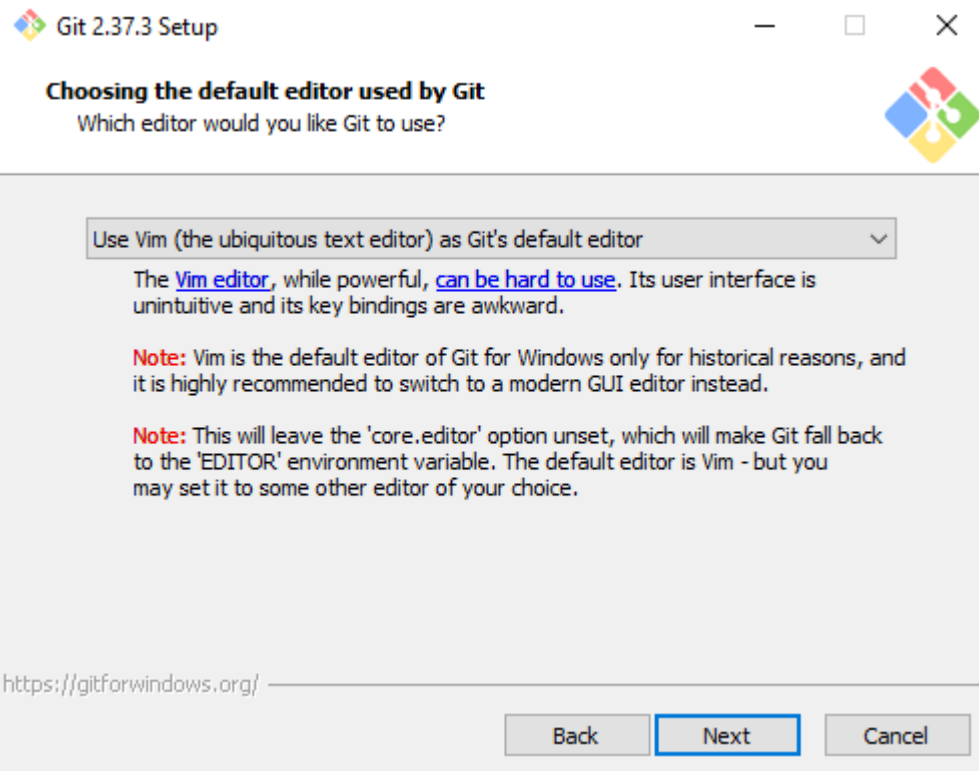
Com o caminho selecionado, clique em "Next" novamente. Selecione todas as caixas de seleção, deixando a última opção sem selecionar. Caso você utilize a ferramenta Windows Terminal, pode selecionar a última opção. Do contrário, deixe ela desmarcada.



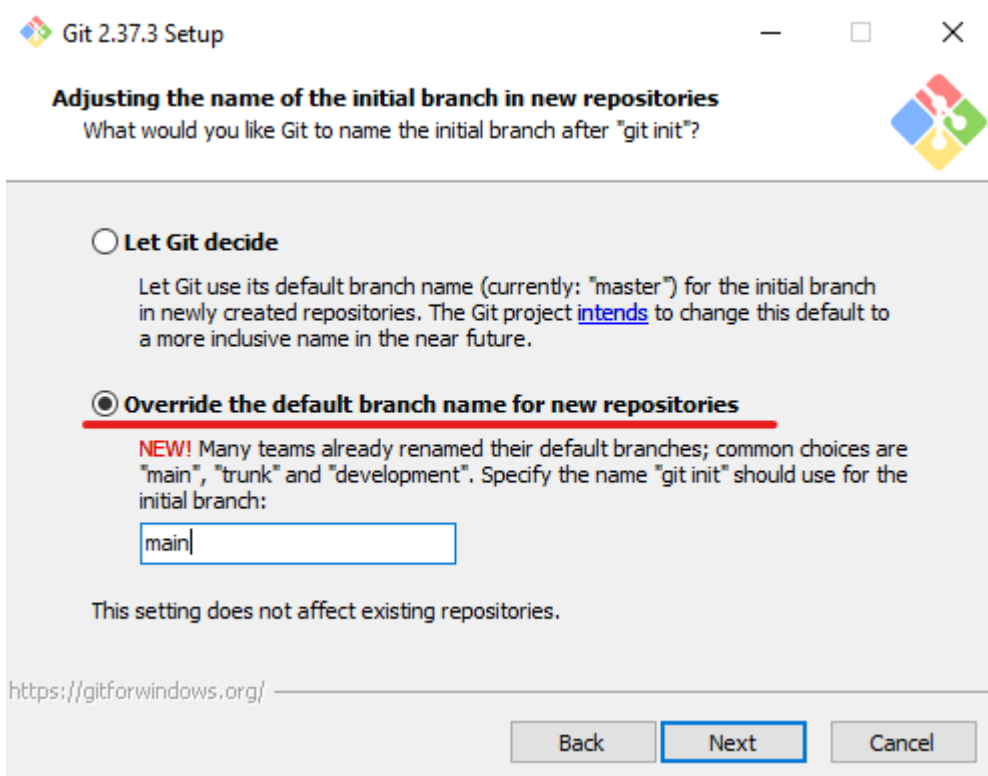
Com as opções selecionadas, clique em "Next" novamente. A próxima tela questiona se desejamos criar um atalho no menu inicial do Windows. Vamos deixar da maneira que está para que o atalho seja criado.



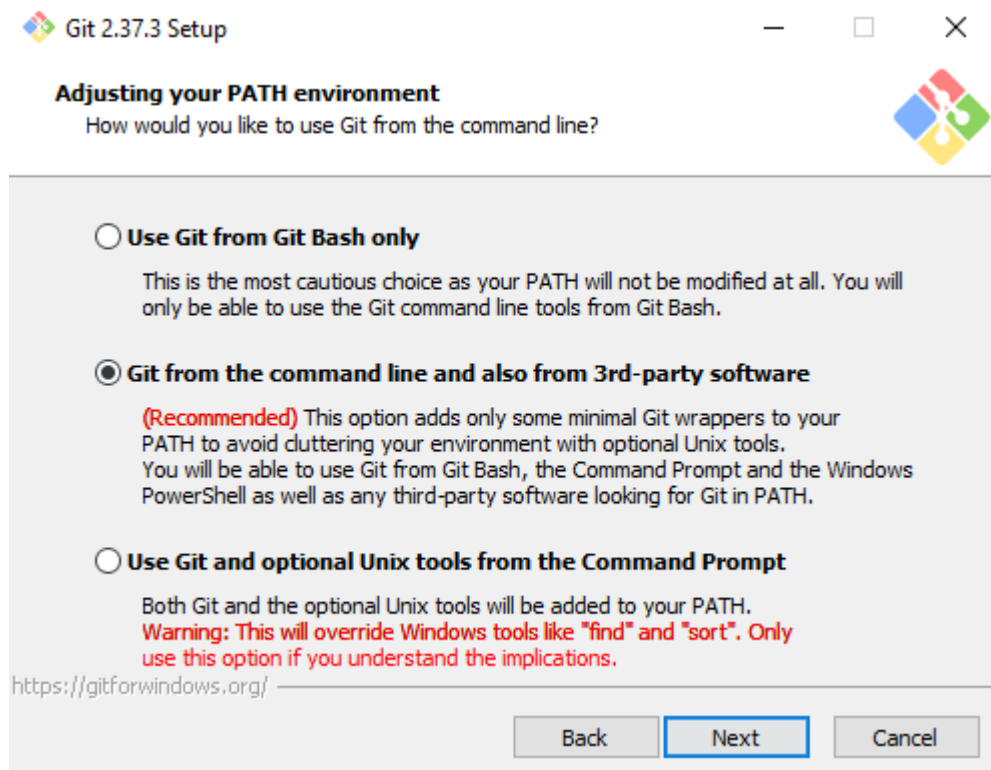
Clique em "Next" novamente. Na próxima tela, o instalador pede para que selecionemos o editor que será utilizado pelo Git, caso seja necessário alterarmos alguma mensagem que o Git gere resolvendo conflitos ou fazendo *merges* (Não se preocupe, veremos o que é um *merge* ao longo do curso). Inicialmente, vamos deixar com o VIM selecionado, conforme figura abaixo. Caso seja necessário, o editor pode ser alterado posteriormente.



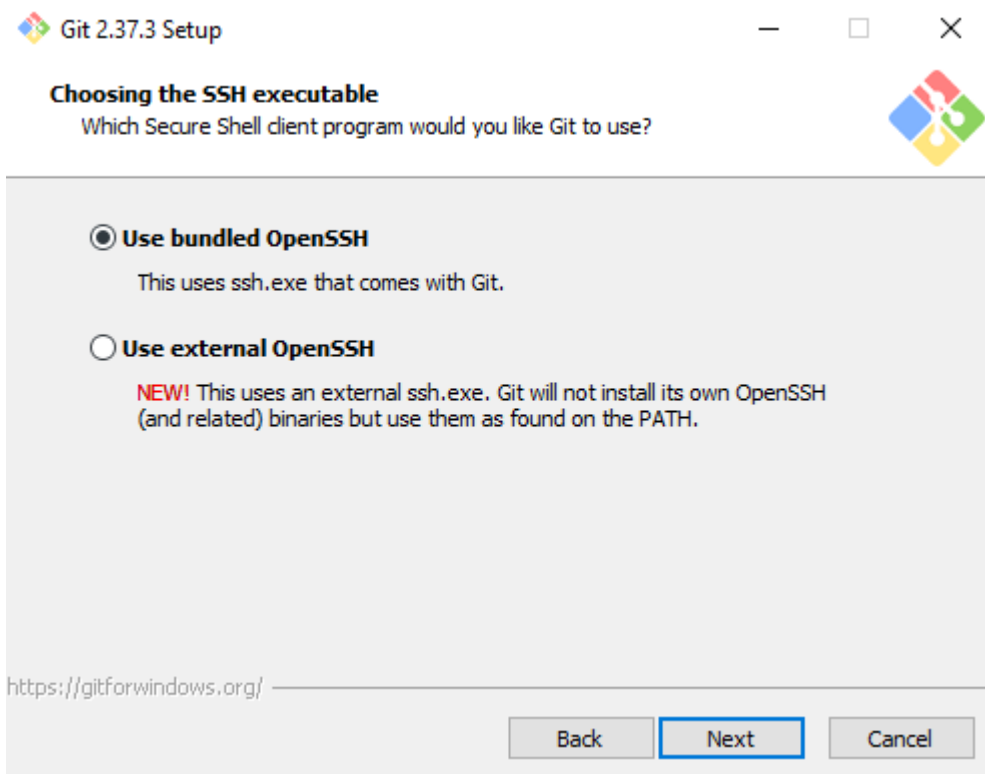
Clique em "Next" novamente. Nesta janela, selecione a opção conforme figura abaixo. Isso irá configurar as nossas *branch* principal como sendo a *main*, e não a *master*. O que vai nos poupar alguns comandos a mais no futuro (novamente, não se preocupe que iremos ver o que é e como trabalhar com *branches* dentro do Git)



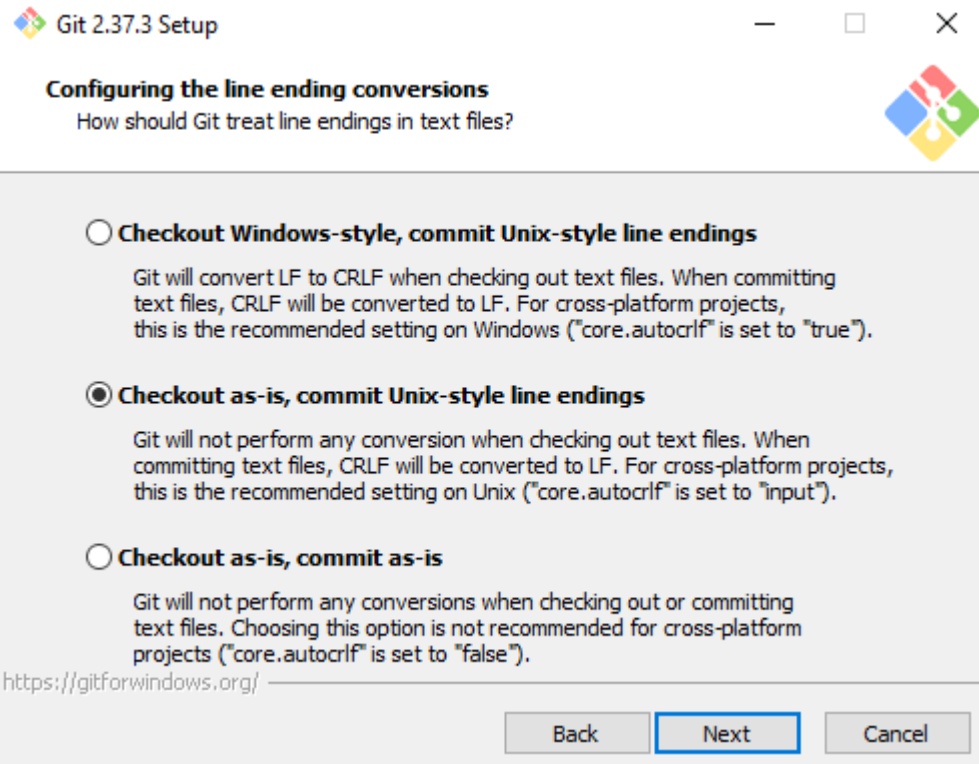
Clique em "Next". Nesta próxima janela, vamos manter a opção padrão.



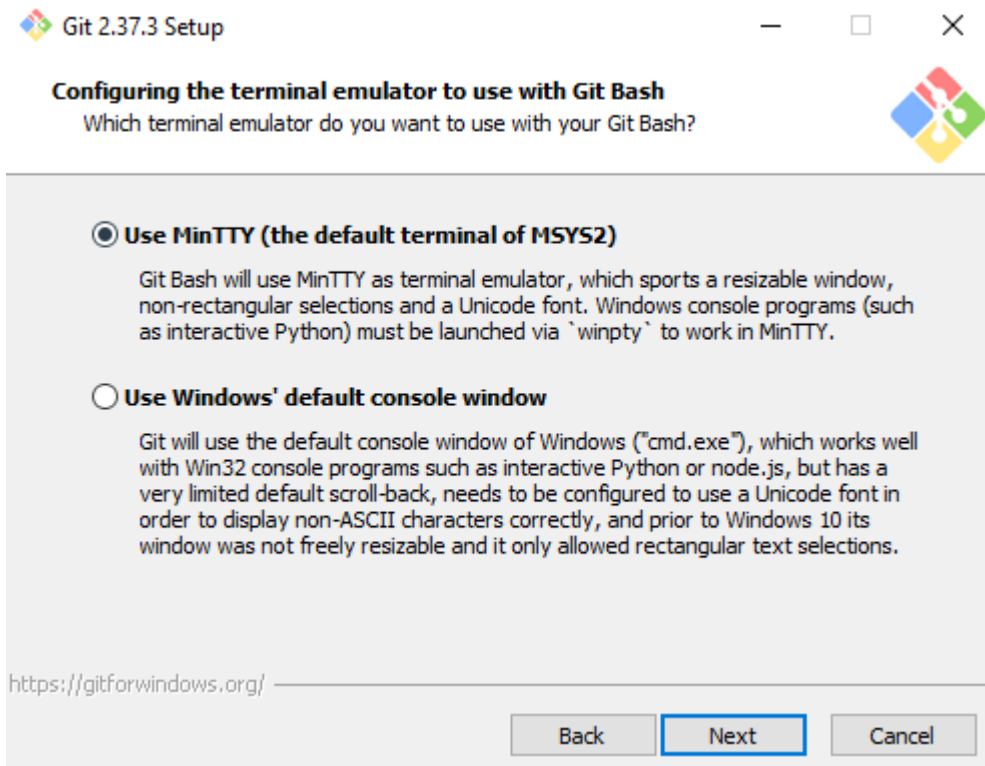
Clique em "Next". Nesta próxima janela, vamos manter a opção padrão.



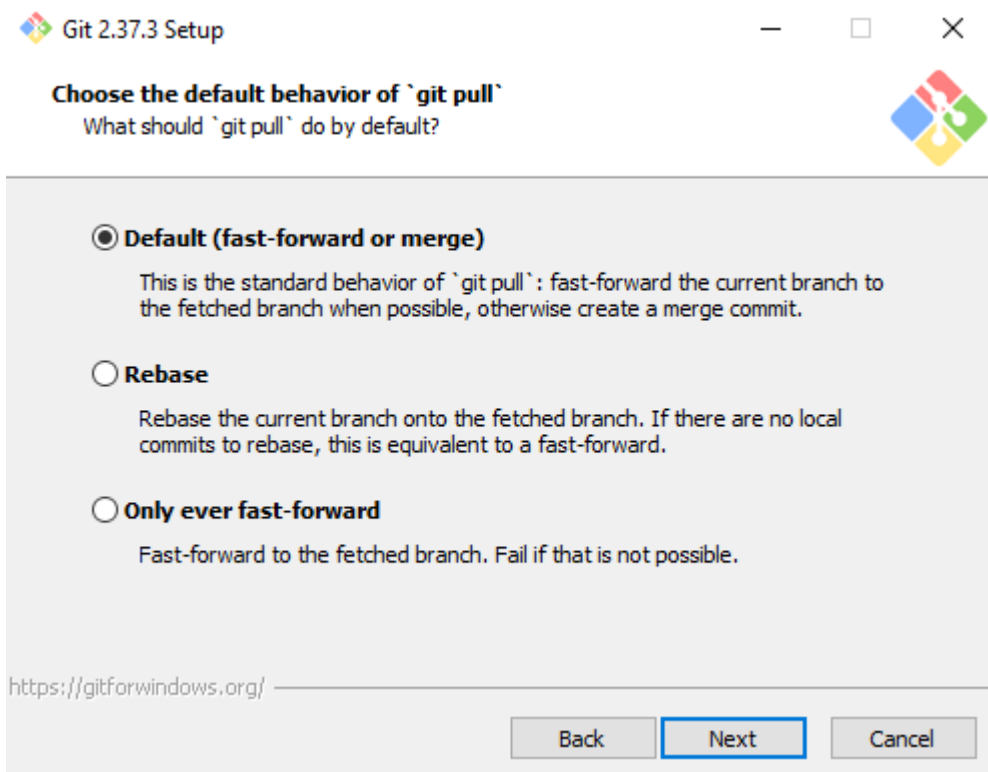
Clique em "Next". Nesta janela, vamos selecionar a opção para alterar os *commits* para o padrão UNIX (Novamente, não se preocupe que iremos abordar o que é um *commit* ao longo do curso). O que essa opção fará é alterar o caractere de final de linha para o padrão UNIX, que é o padrão utilizado em servidores.



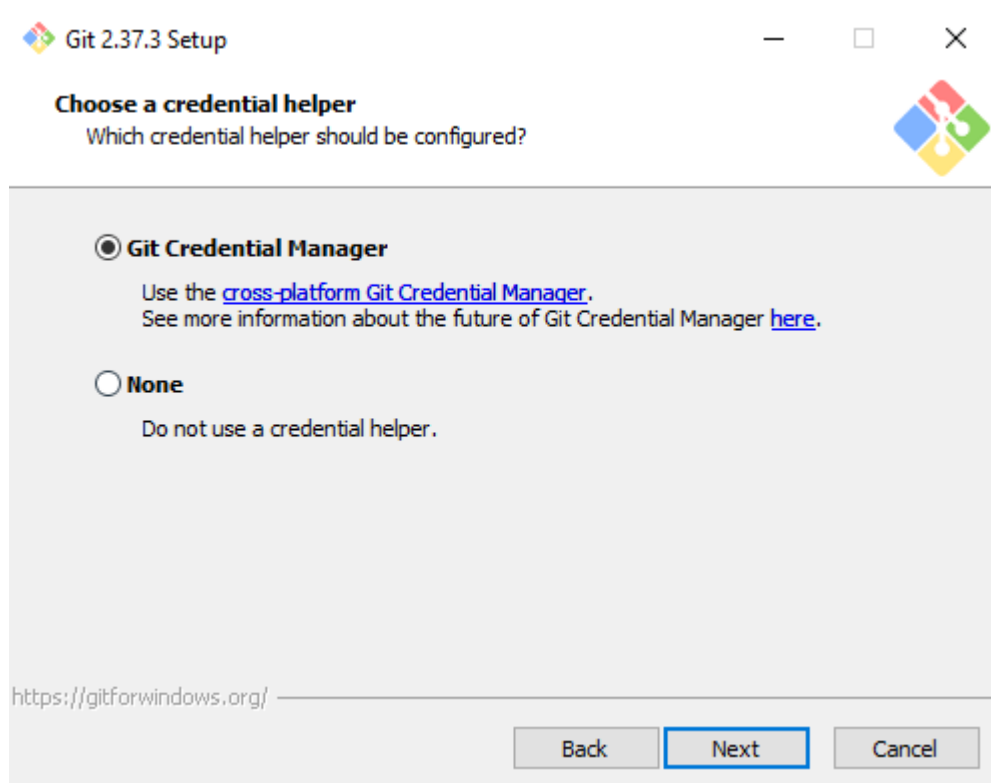
Clique em "Next". Nesta próxima janela, vamos manter a opção padrão.



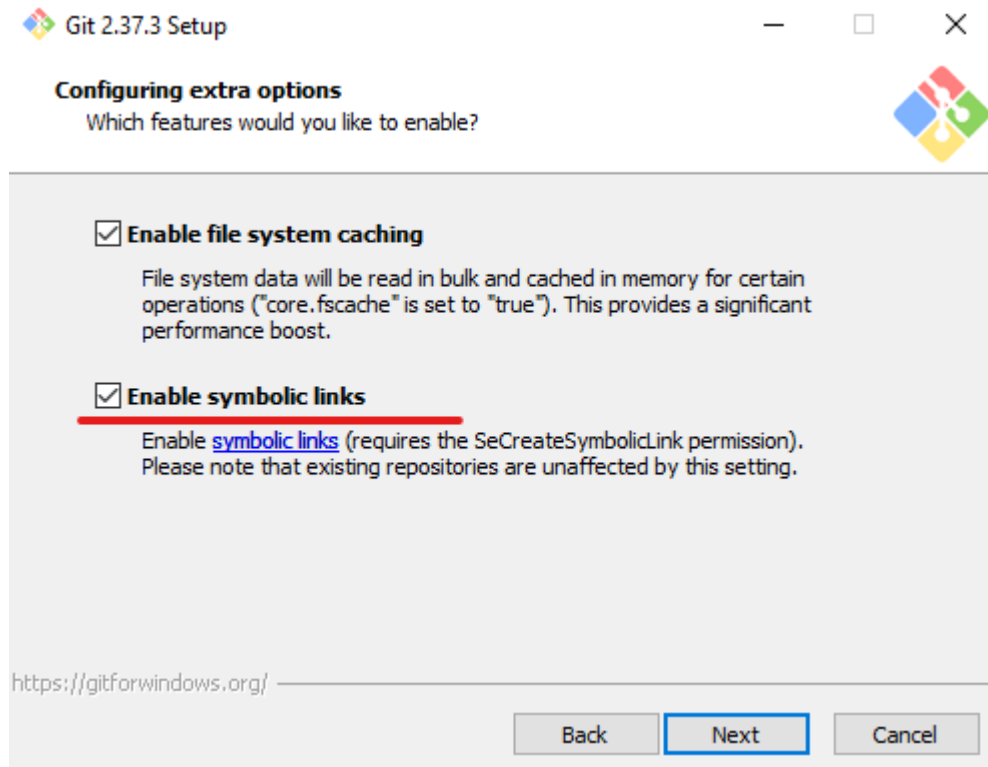
Clique em "Next". Nesta próxima janela, vamos manter a opção padrão.



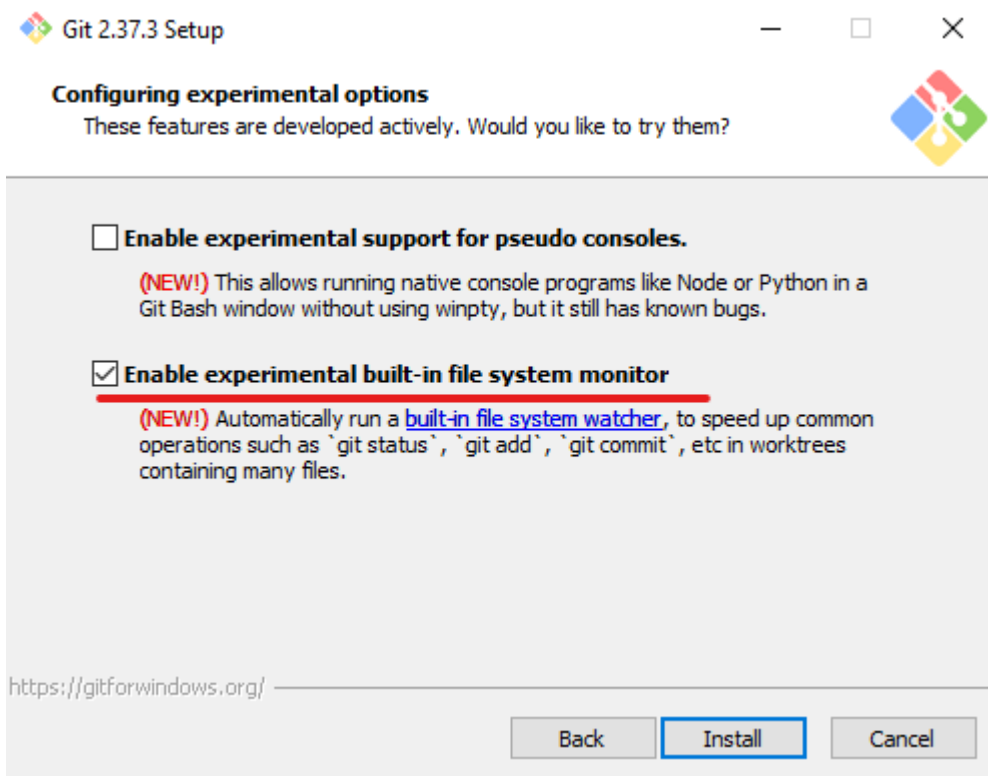
Clique em "Next". Essa tela nos pergunta se desejamos utilizar o [Git Credential Manager](#) para armazenarmos as nossas credenciais. Vamos manter essa opção selecionada.



Clique em "Next". Nesta janela, vamos selecionar a opção de habilitar a criação de links simbólicos no Windows. Embora seja uma opção não utilizaremos no curso, pode vir a ser útil ao longo dos seus projetos.



Clique em "Next" novamente. Por fim, selecione a opção que habilita o monitor de sistema de arquivos. Essa opção deixará alguns comandos mais performáticos para repositórios com muitos arquivos, ao troco de um pouco de processamento.



Feito isso, basta clicar no botão install e aguardar o fim da instalação. Ao final da instalação, basta abrir a aplicação "Git Bash" para iniciarmos com o processo de configuração inicial.

Configurações Iniciais

Com o Git instalado, basta fazer as configurações iniciais. Para isso, com o Git Bash aberto basta digitar o comando:

```
$ git config --global user.name "Nome"
```

Para configurar o nome do seu usuário de forma global, e o comando:

```
$ git config --global user.email johndoe@example.com
```

Para configurar o email do seu usuário de forma global. Feito isso, para verificar se as configurações foram feitas corretamente, basta digitar o comando:

```
$ git config --list
```

Para exibir todas as configurações feitas dentro do Git.

Próxima Aula

Na próxima aula, iremos fazer o download que será utilizado ao longo do curso e iremos preparar o ambiente para conseguirmos fazer as alterações no projeto e também para conseguirmos utilizar os comandos do Git.

Fontes e Links Complementares

[Página Principal do Git](#)

[Documentação - Configurações Iniciais](#)

[Documentação - Alterar o editor de texto do Git](#)

[Documentação - Git Credential Manager](#)