











git e github: conceitos básicos

```
$ echo `whoami`
```

```
Narumi Abe
```

Visão geral

Isso já aconteceu com vocês?

untitled folder				+
Name	^	Size	Kind	
 trabalho (1) copy.docx		193 KB	Micros...(.docx)	
 trabalho (1) versao final.docx		193 KB	Micros...(.docx)	
 trabalho (1).docx		193 KB	Micros...(.docx)	
 trabalho com correções.docx		193 KB	Micros...(.docx)	
 trabalho final final.docx		193 KB	Micros...(.docx)	
 trabalho final.docx		193 KB	Micros...(.docx)	
 trabalho ultimo final 2.docx		193 KB	Micros...(.docx)	
 trabalho ultimo final mesmo.docx		193 KB	Micros...(.docx)	
 trabalho ultimo final.docx		193 KB	Micros...(.docx)	
 trabalho.docx		193 KB	Micros...(.docx)	

□ que é git?

- Criado por Linus Torvalds
- Sistema de controle de versão para códigos-fontes
- Plataforma colaborativa
- Software livre



git

□ que é github?

Site e rede social que hospeda softwares livres.
Usa o git para manutenção do código.



github != git

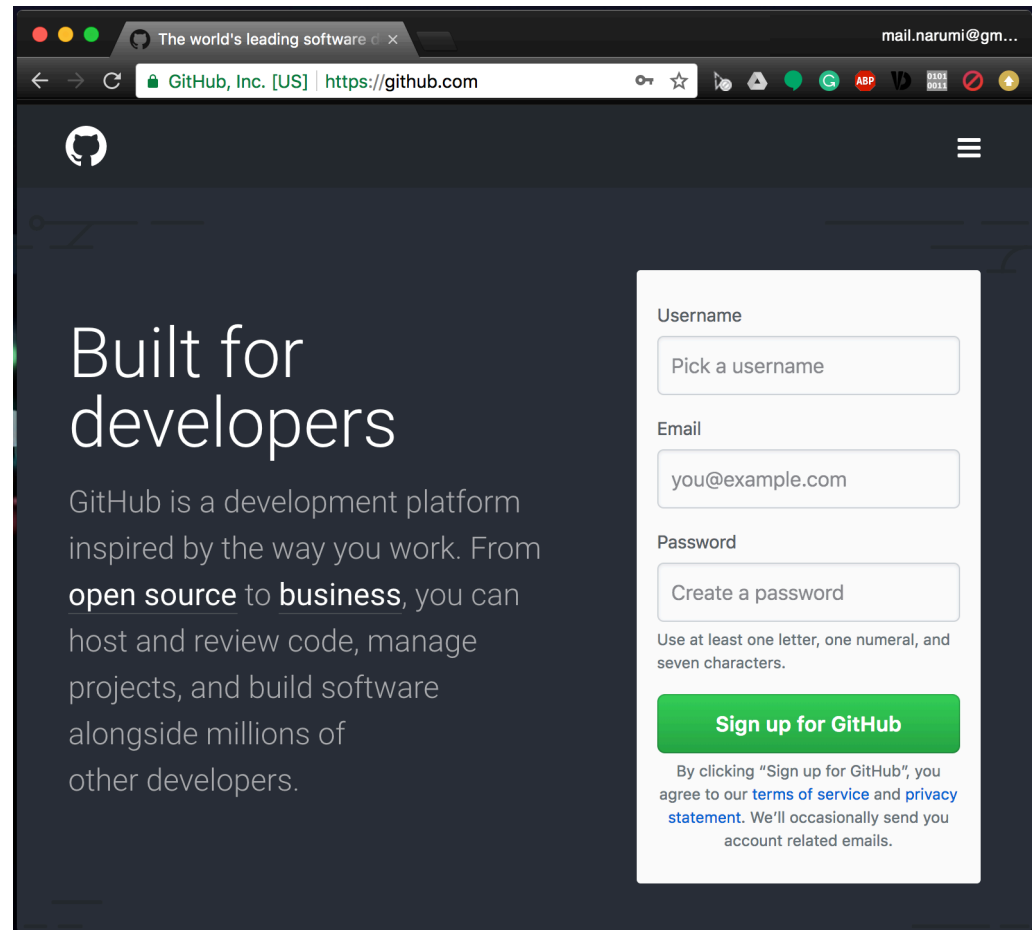
O github é só um site que utiliza a tecnologia git.
Existem diversos sites semelhantes.



Criando uma conta no **github**

Comece cadastrando um usuário e confirme o seu email como em qualquer site.

<https://github.com>



The screenshot shows the GitHub website in a web browser. The browser's address bar displays "https://github.com". The page features the GitHub logo and a navigation menu. The main content area has the heading "Built for developers" and a description of GitHub as a development platform. On the right side, there is a sign-up form with the following fields and elements:

- Username:** A text input field with the placeholder "Pick a username".
- Email:** A text input field with the placeholder "you@example.com".
- Password:** A text input field with the placeholder "Create a password".
- Instructions:** Below the password field, it says "Use at least one letter, one numeral, and seven characters."
- Sign up button:** A green button labeled "Sign up for GitHub".
- Terms and Privacy:** Below the button, it states: "By clicking 'Sign up for GitHub', you agree to our [terms of service](#) and [privacy statement](#). We'll occasionally send you account related emails."

Instalando o git – Unix

Mac

```
$ brew install git
```

Ubuntu

```
$ apt install git
```

Fedora

```
$ yum install git
```

Suse

```
$ zypper install git
```

FreeBSD

```
$ pkg install git
```

Instalando o git – Windows

1. Abra o site

<https://git-scm.com/>

2. Clique em download,

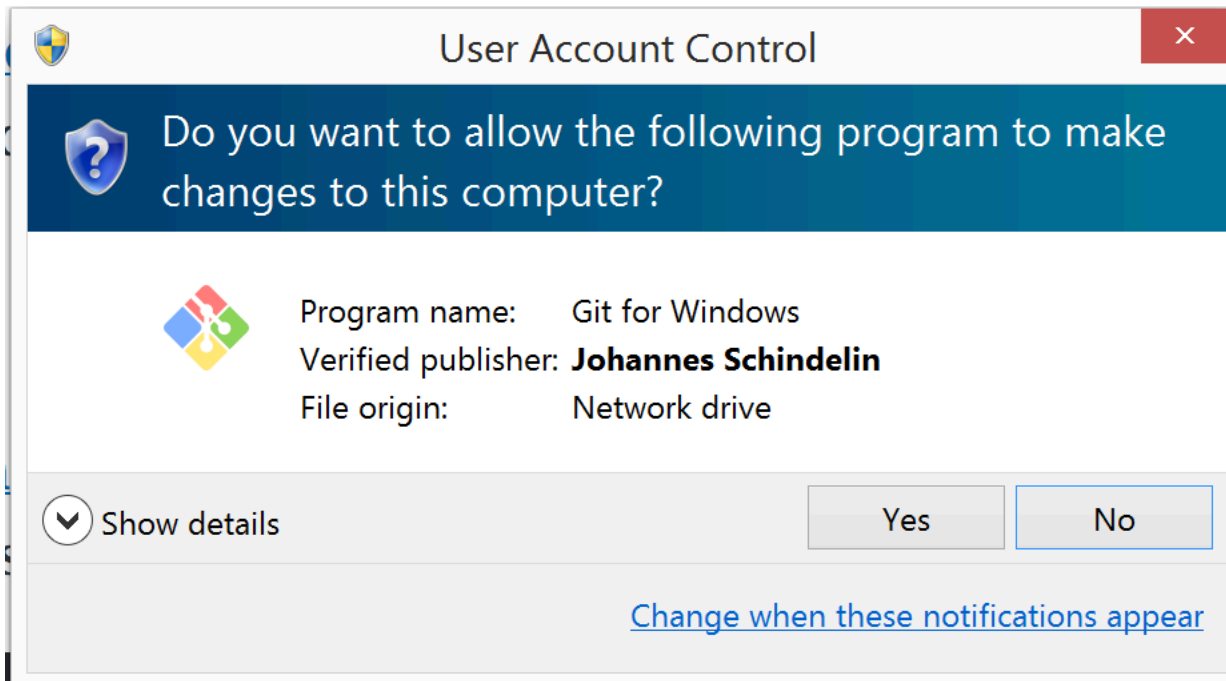
3. Clique em windows.

4. Clique em baixar o arquivo.

5. Abra o arquivo baixado

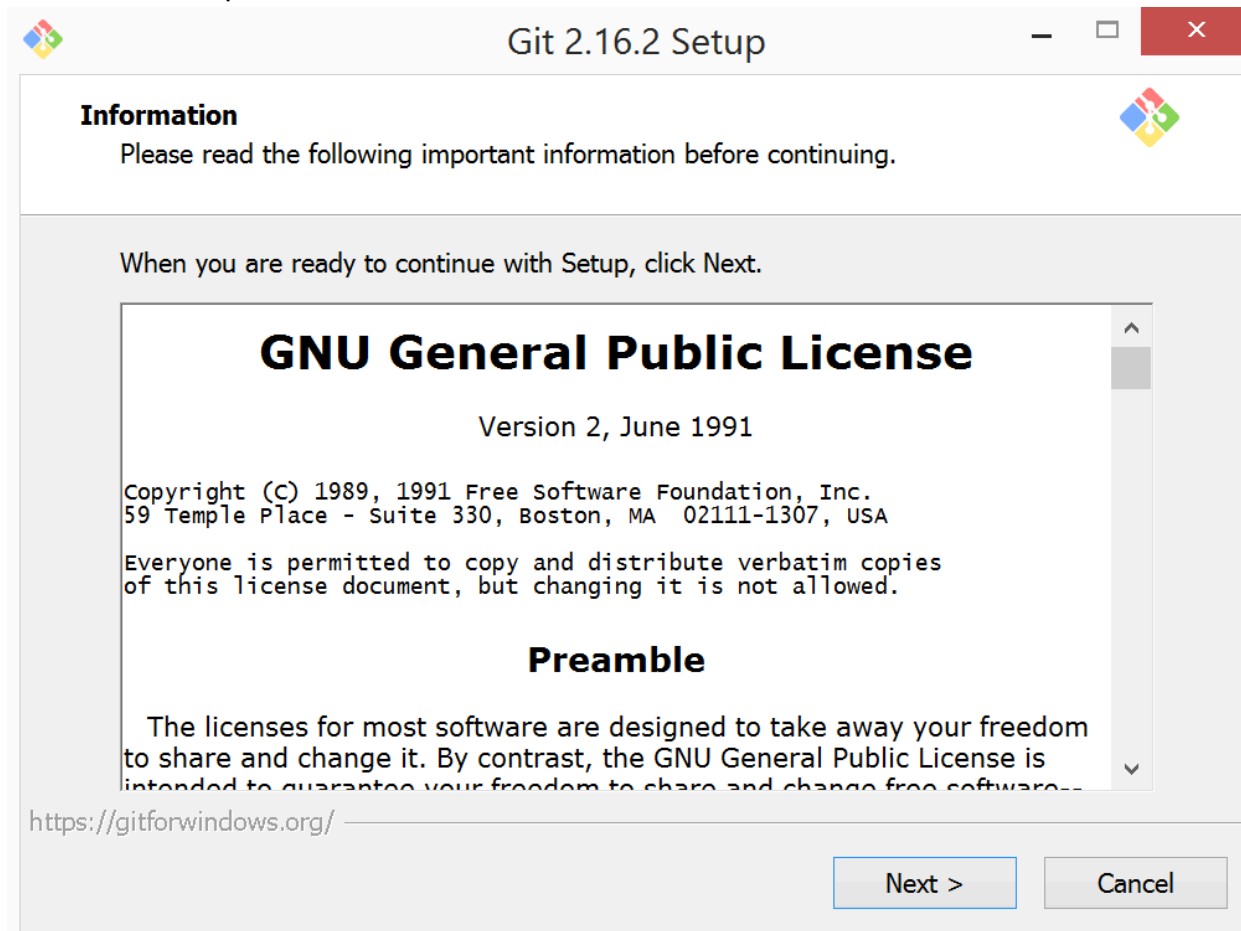
Instalando o git – Windows

6. Aperte OK na mensagem de segurança do windows



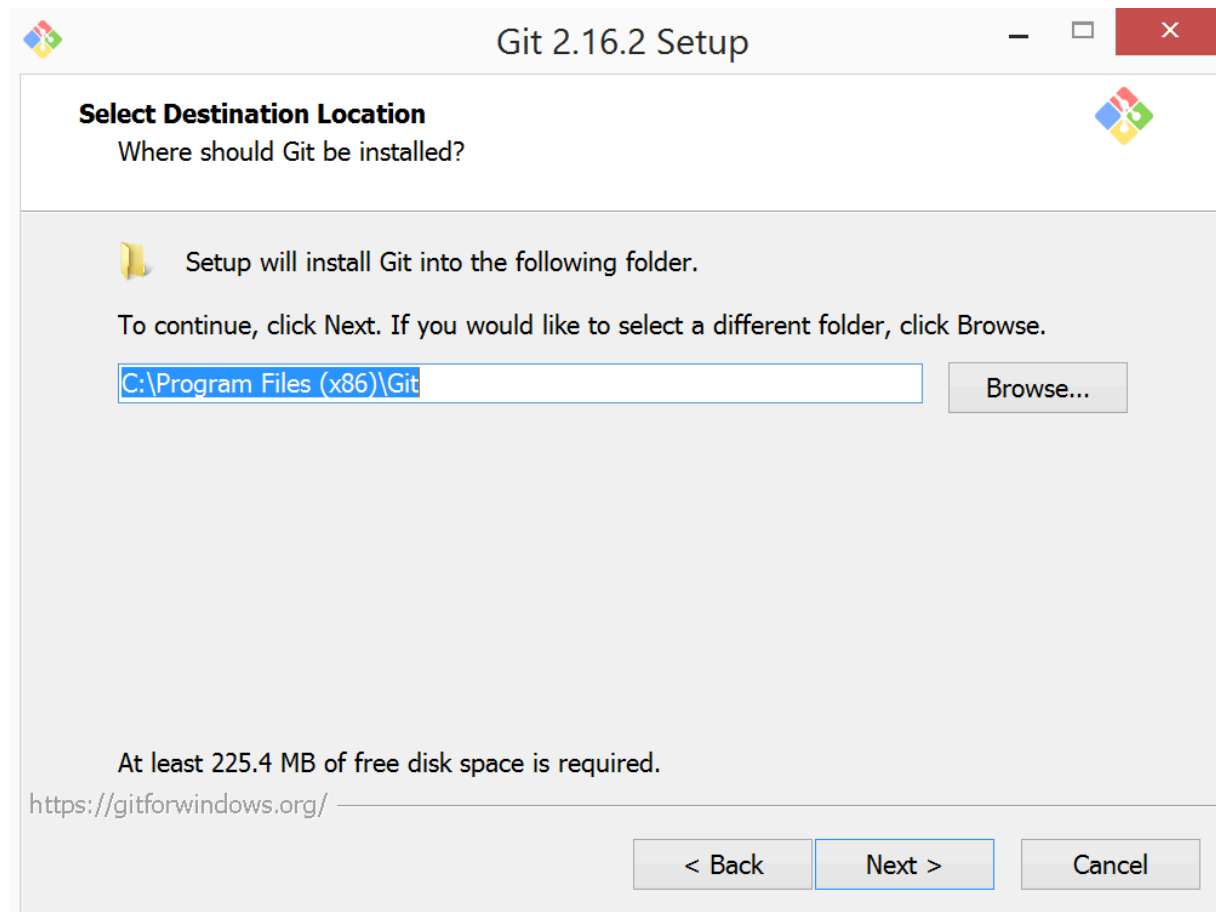
Instalando o git – Windows

7. Aperte next para a licença



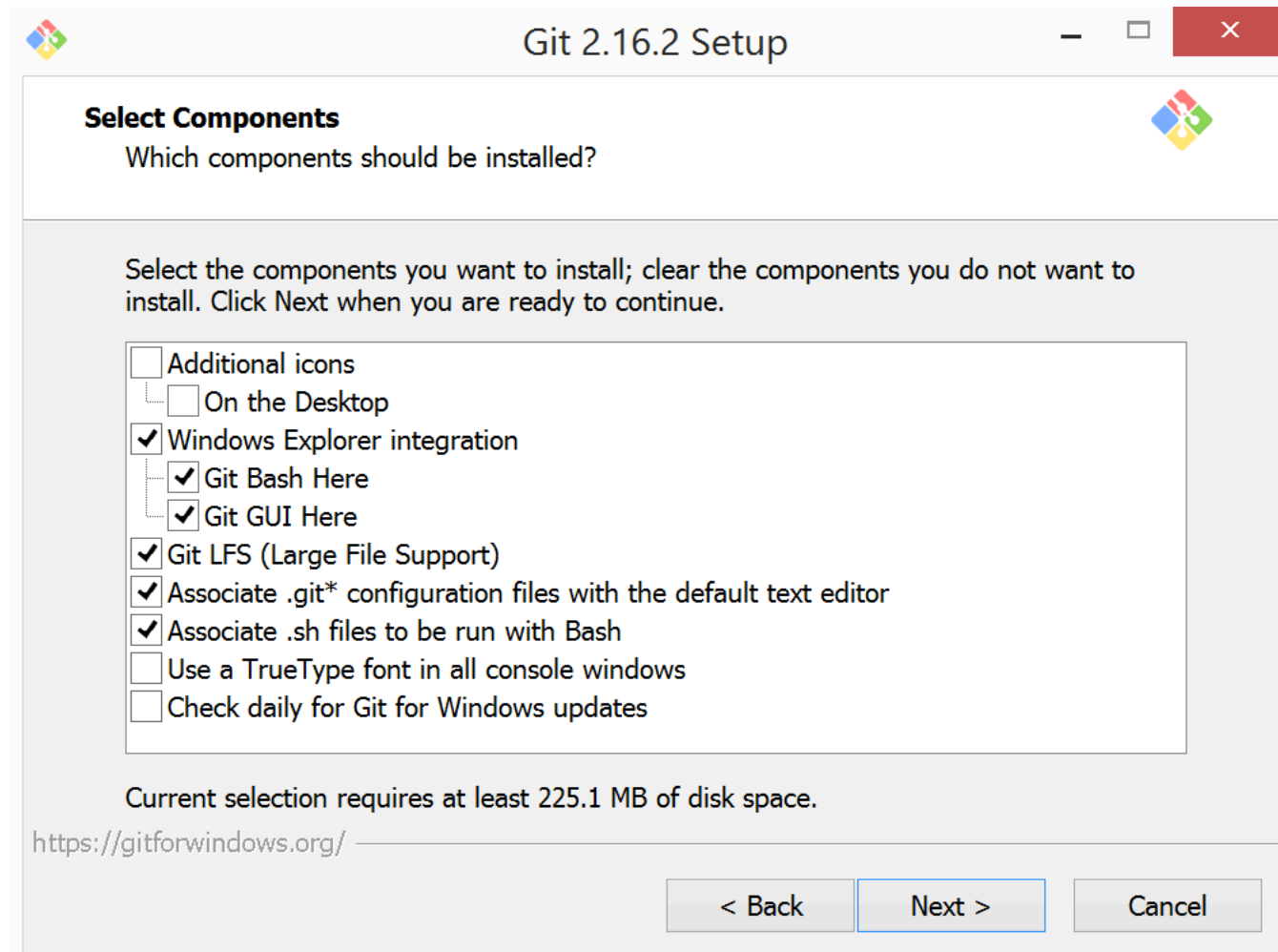
Instalando o git – Windows

8. Escolha o local da instalação



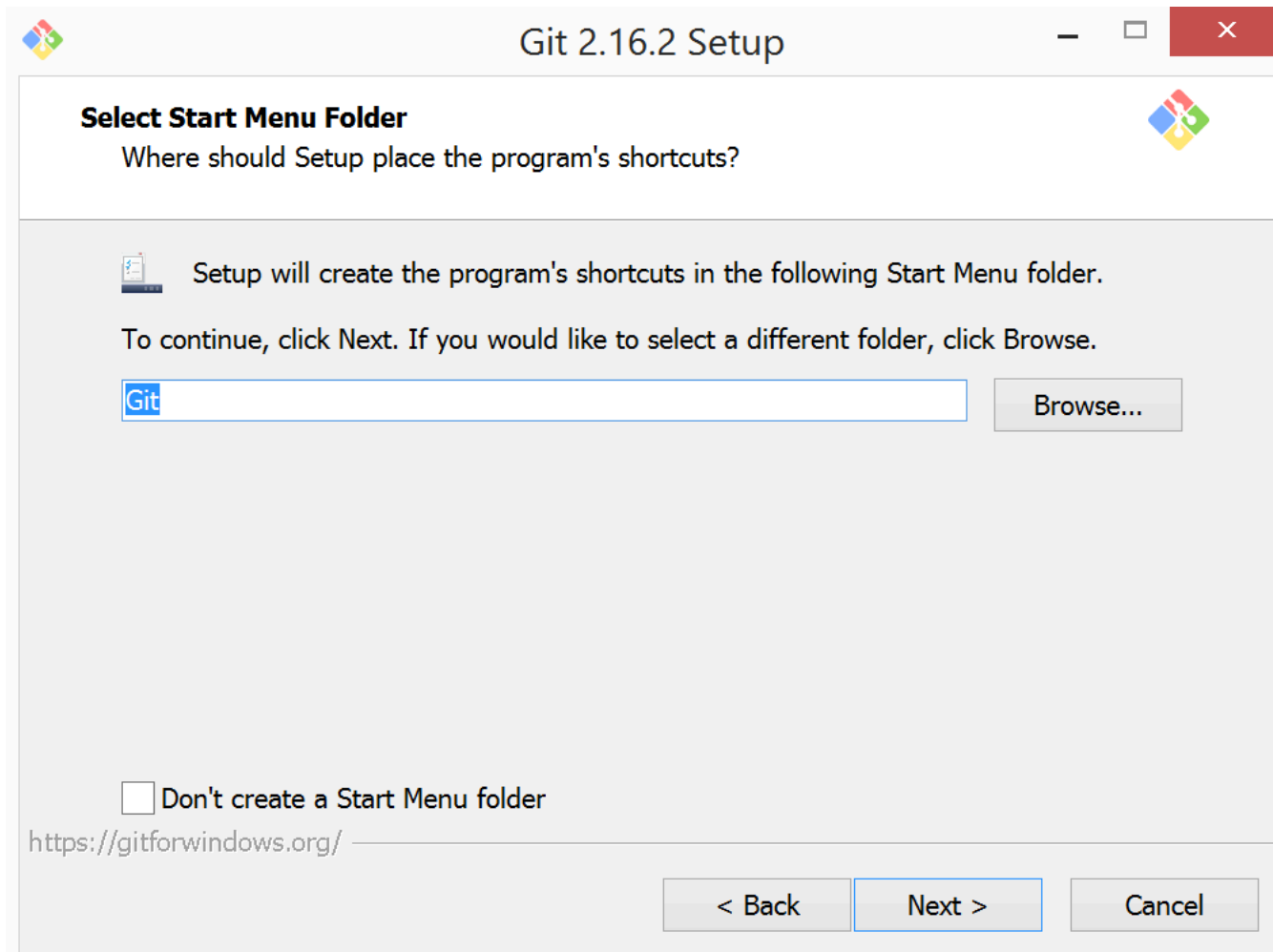
Instalando o git – Windows

9. Aperte next



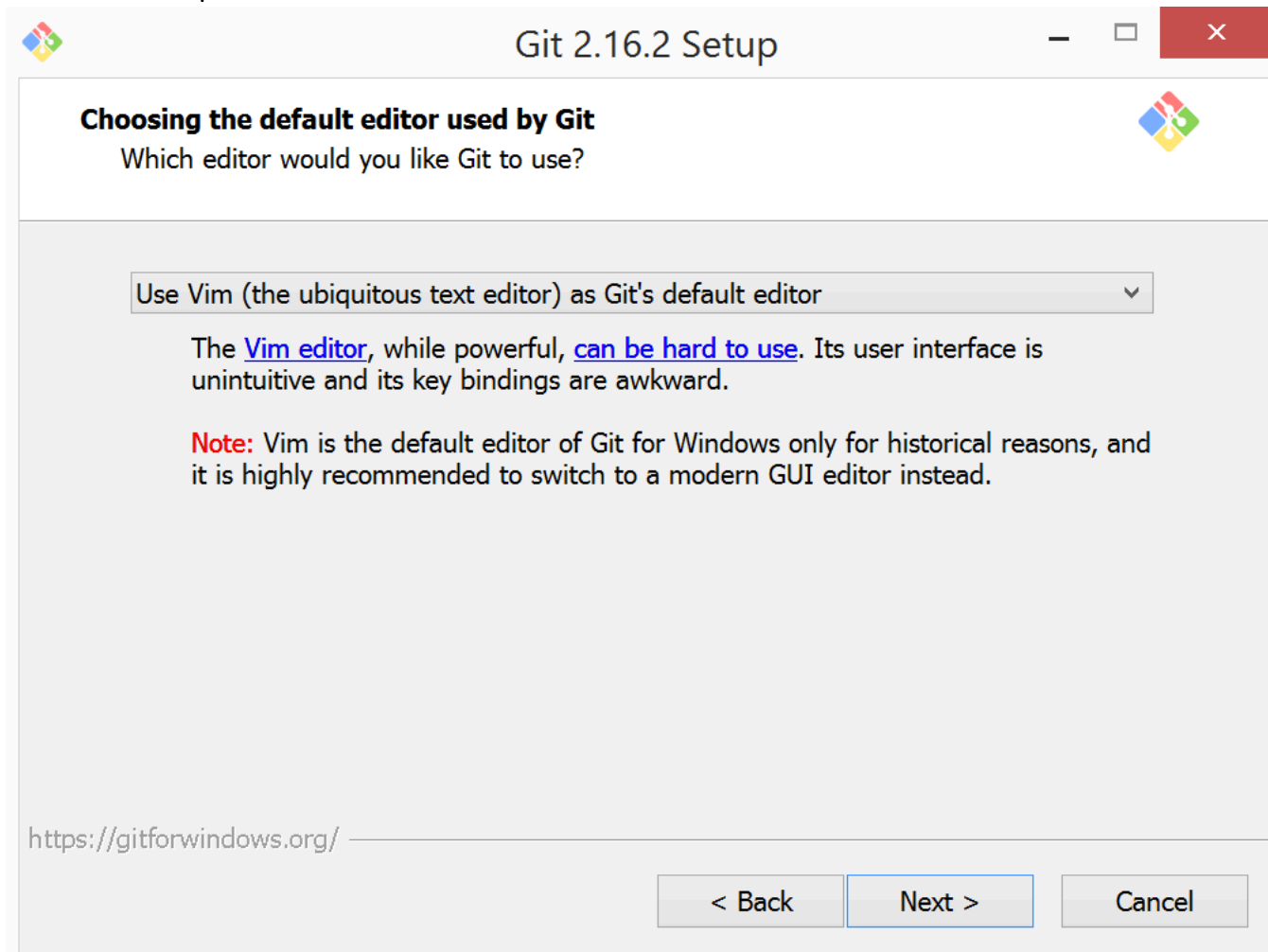
Instalando o git – Windows

10. Só mais alguns nexts...



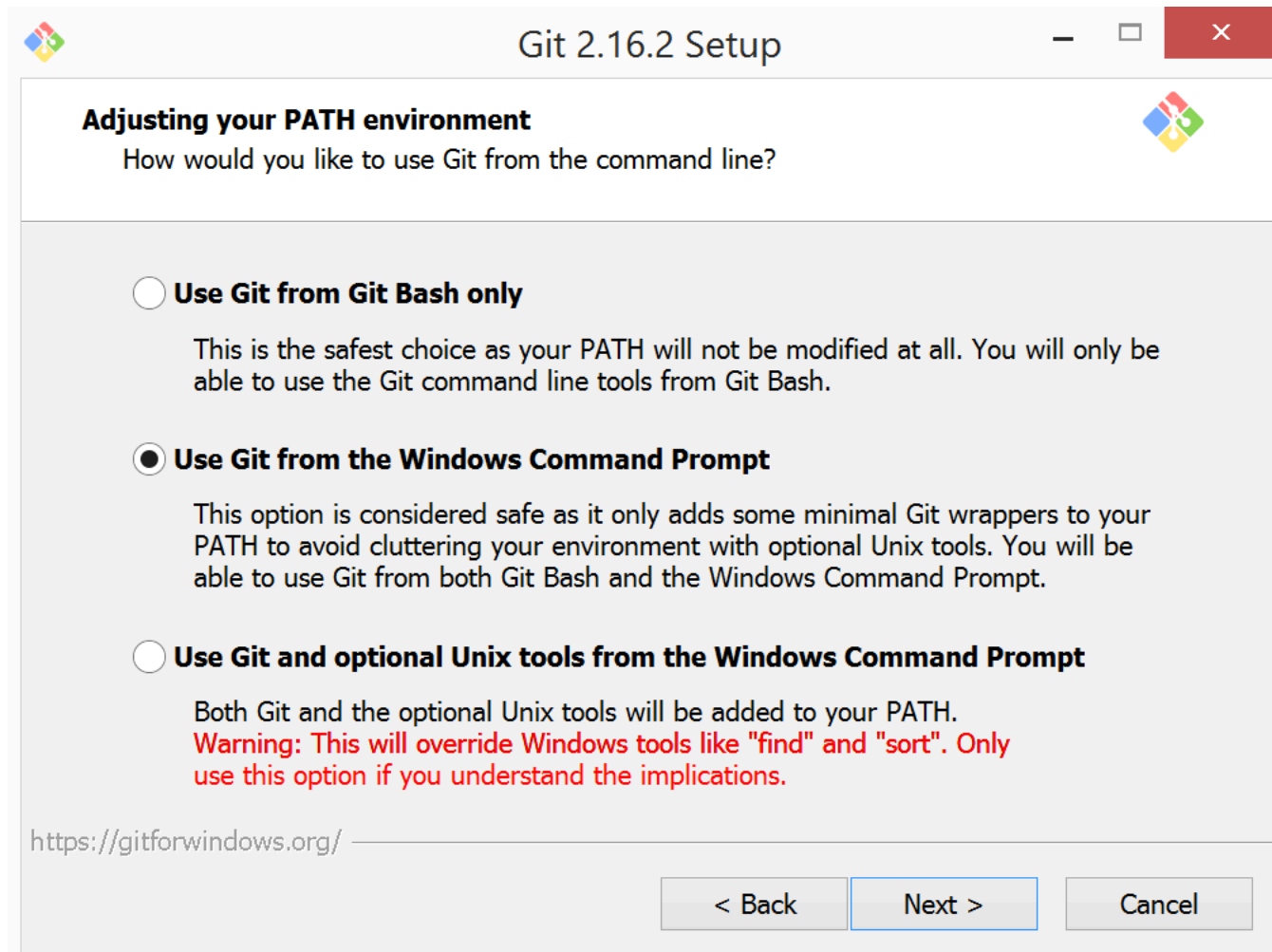
Instalando o git – Windows

11. Deixe a opção Vim selecionada



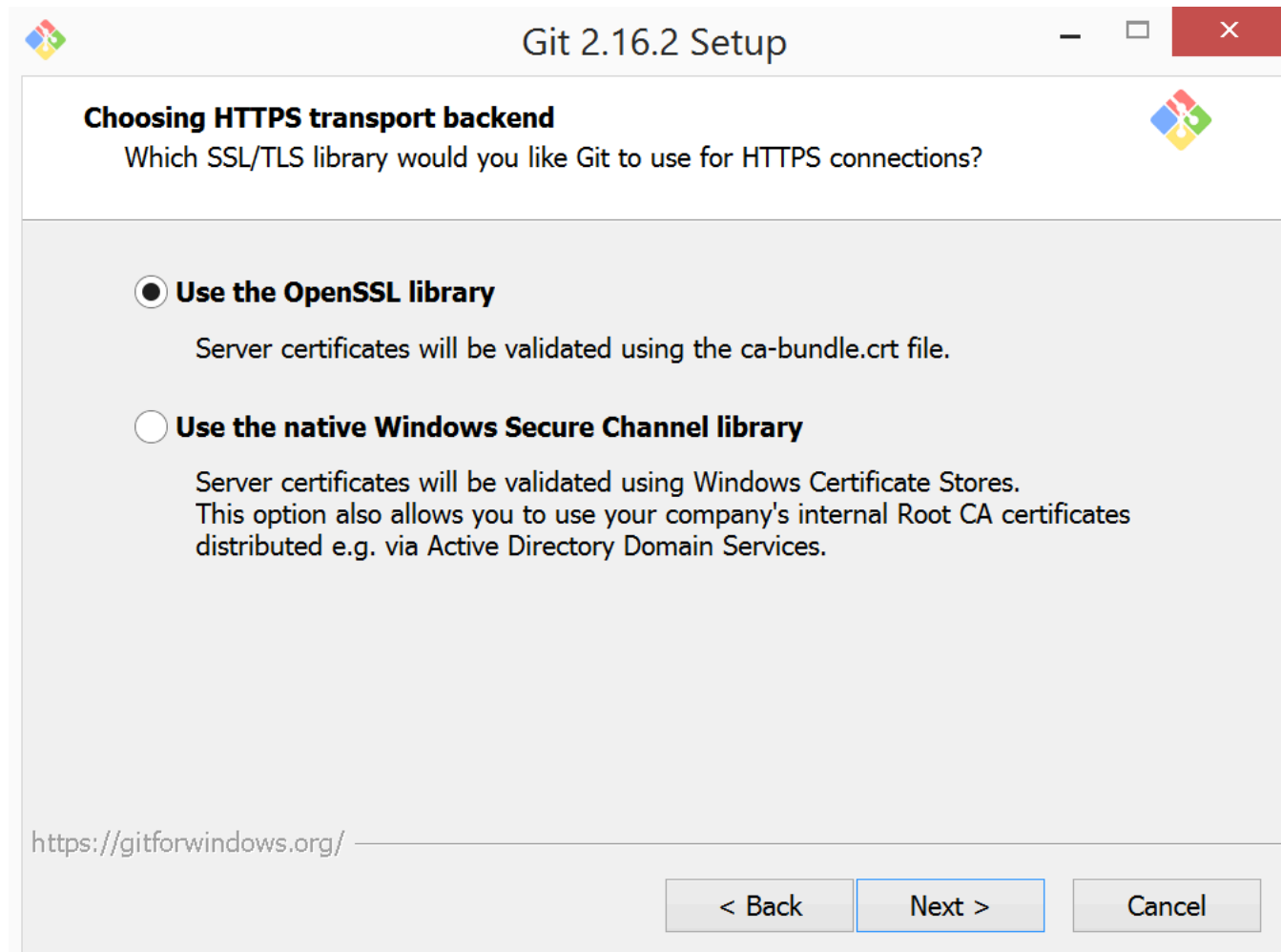
Instalando o git – Windows

12. Next



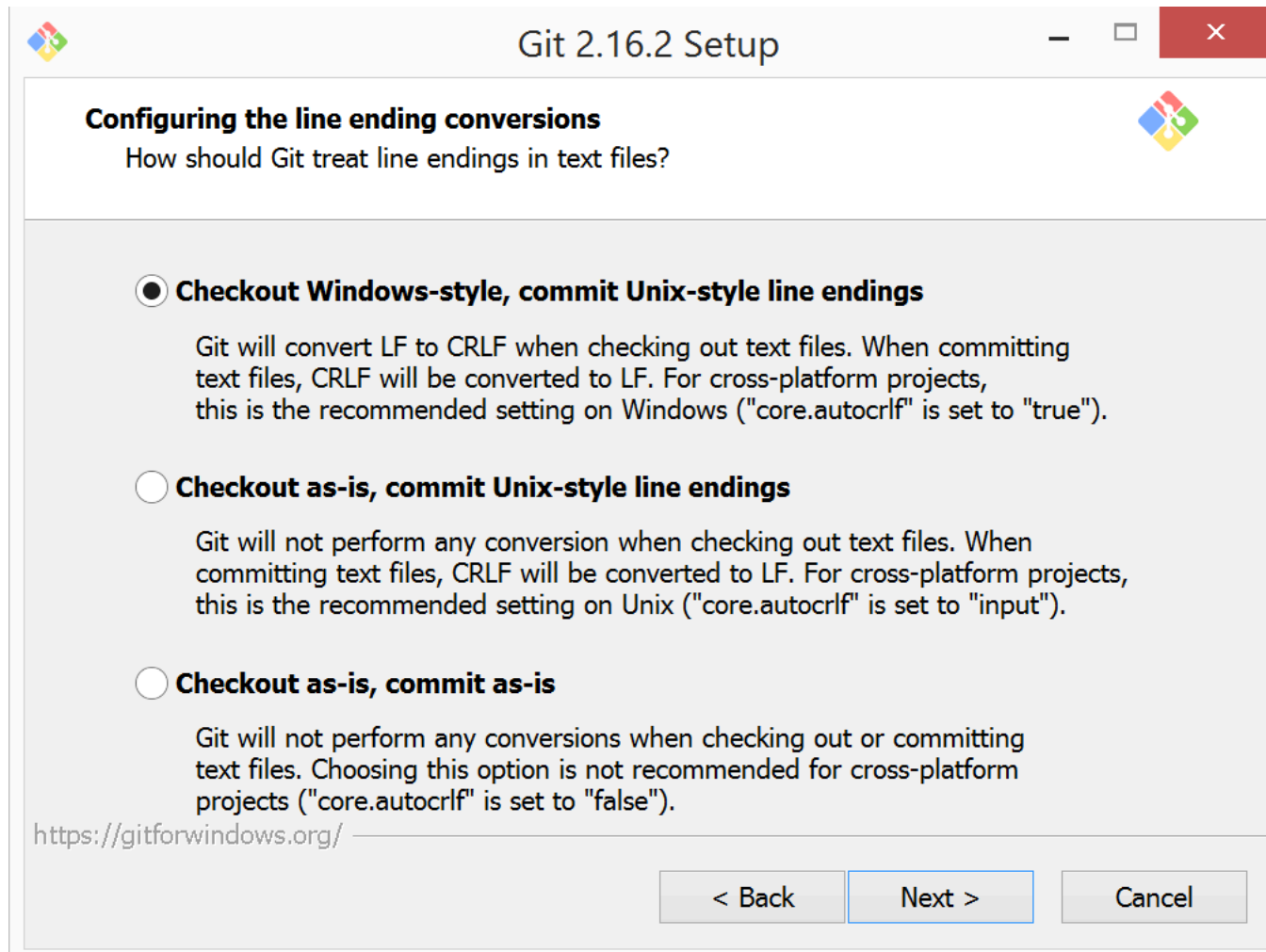
Instalando o git – Windows

13. Next



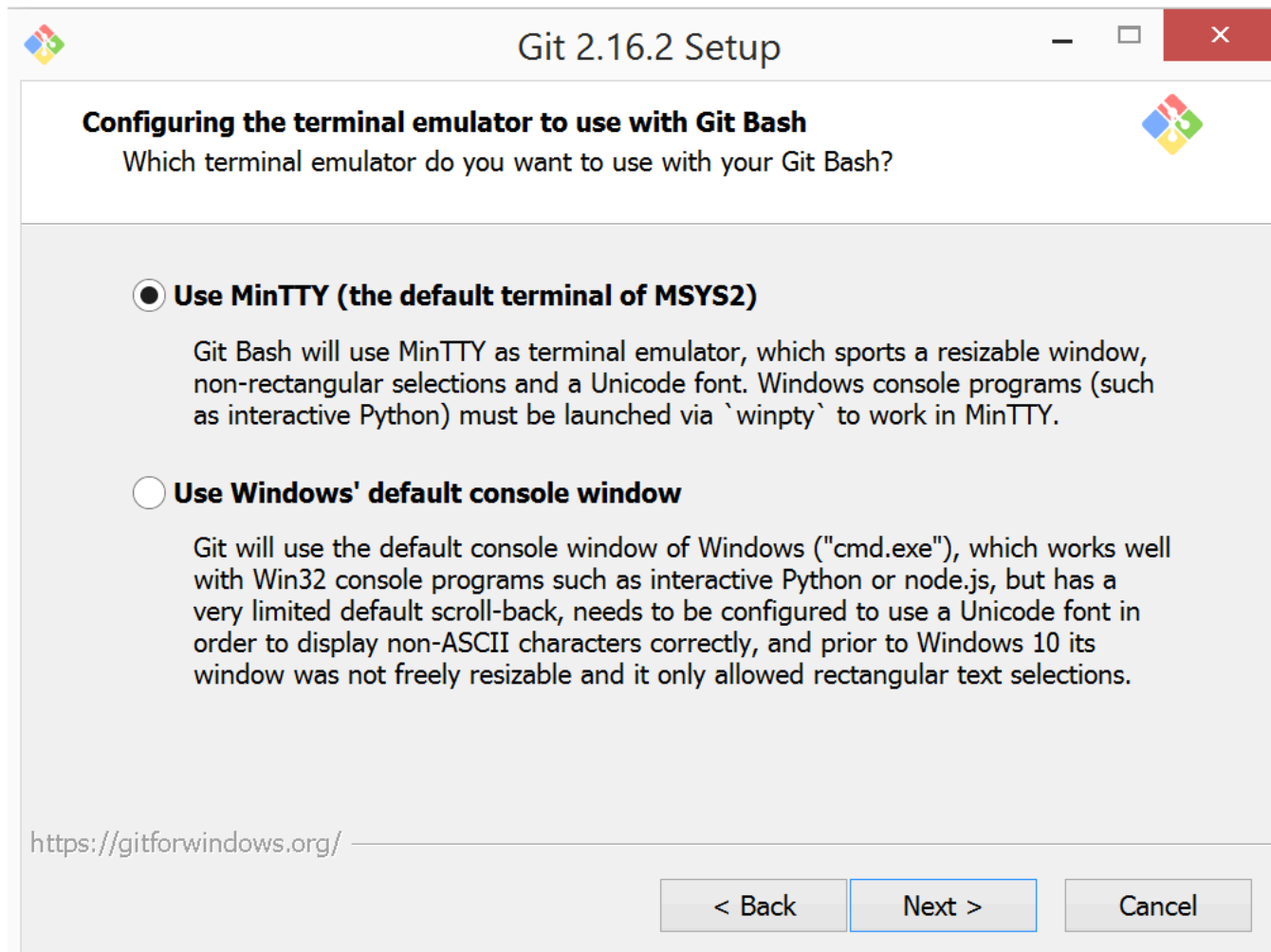
Instalando o git – Windows

14. Next



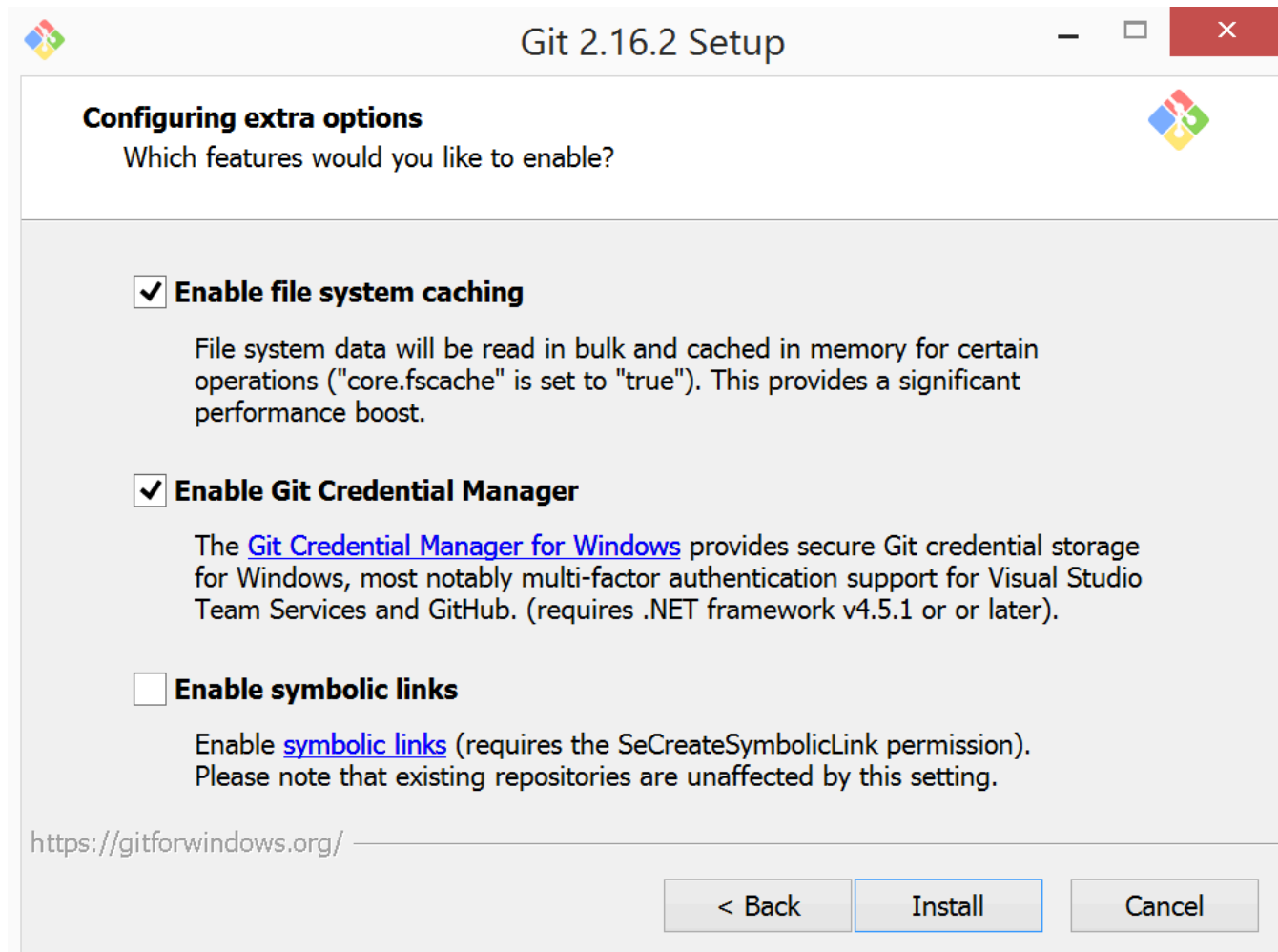
Instalando o git – Windows

15. Next



Instalando o git – Windows

16. Clique em Install e reinicie o computador (possivelmente)



Conceitos chave

É imprescindível entendermos os conceitos e usarmos o mesmo vocabulário do git

Conceitos chave

repo

commit

branch

pull request

Repositório

Repositório

O repositório é o endereço de um projeto. Consiste em um ou mais arquivos que contêm os código-fontes de um sistema (software, página, app) em alguma linguagem de programação.



This repository

Search

Pull requests

Issues

Marketplace

Explore



python / cpython

Watch

829

★ Star

16,801

Fork

4,787

<> Code

Pull requests 624

Insights

The Python programming language <https://www.python.org/>

101,206 commits

7 branches

349 releases

509 contributors

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download



ericvsmith Minor fixes to dataclass tests. (GH-6243) ...

Latest commit 2b75fc2 an hour ago

github	Update PR template file, don't reference bpo 12345 (GH-5897)	28 days ago
Doc	Clarify fd inheritance when close_fds=False. (GH-6240)	8 hours ago
Grammar	bpo-30406: Make async and await proper keywords (#1669)	6 months ago
Include	bpo-33041: Rework compiling an "async for" loop. (#6142)	3 days ago
Lib	Minor fixes to dataclass tests. (GH-6243)	an hour ago
Mac	Add macOS installer Conclusion file	13 days ago
Misc	bpo-33053: -m now adds *starting* directory to sys.path (GH-6231)	12 hours ago
Modules	bpo-33136: Harden ssl module against CVE-2018-8970 (GH-6229)	15 hours ago
Objects	bpo-30953: Improve error messages and add tests for jumping (GH-6196)	3 days ago

O Python é uma das linguagens mais utilizadas e poderosas atualmente.
Seu código é todo controlado e mantido em um repositório no github.

Commit

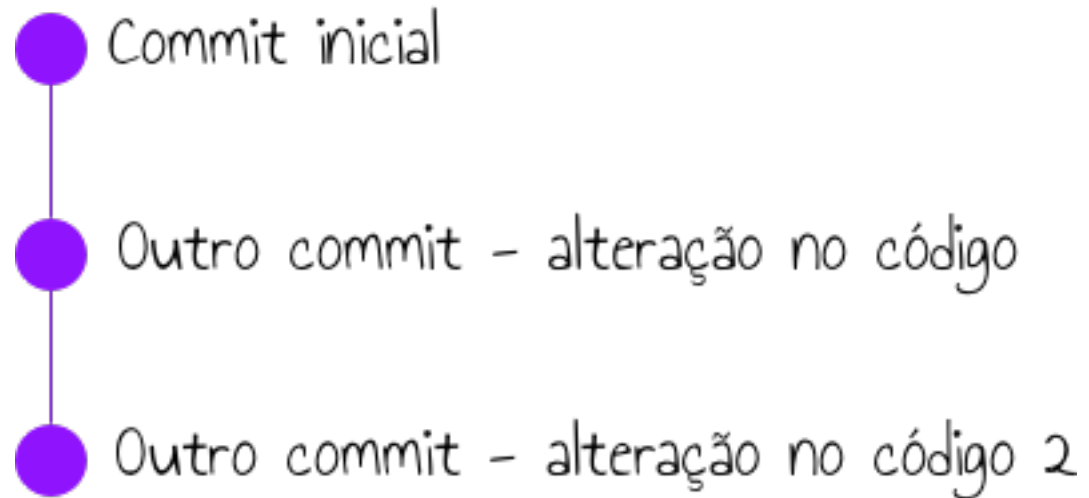
commit

O **commit** é o processo de avisar as alterações feitas no código fonte de um repositório. **Por hora**, vamos pensar no **commit** como um sinônimo de **salvar**.

Sempre que fazemos alguma alteração no código, precisamos fazer um commit para que as alterações sejam confirmadas.

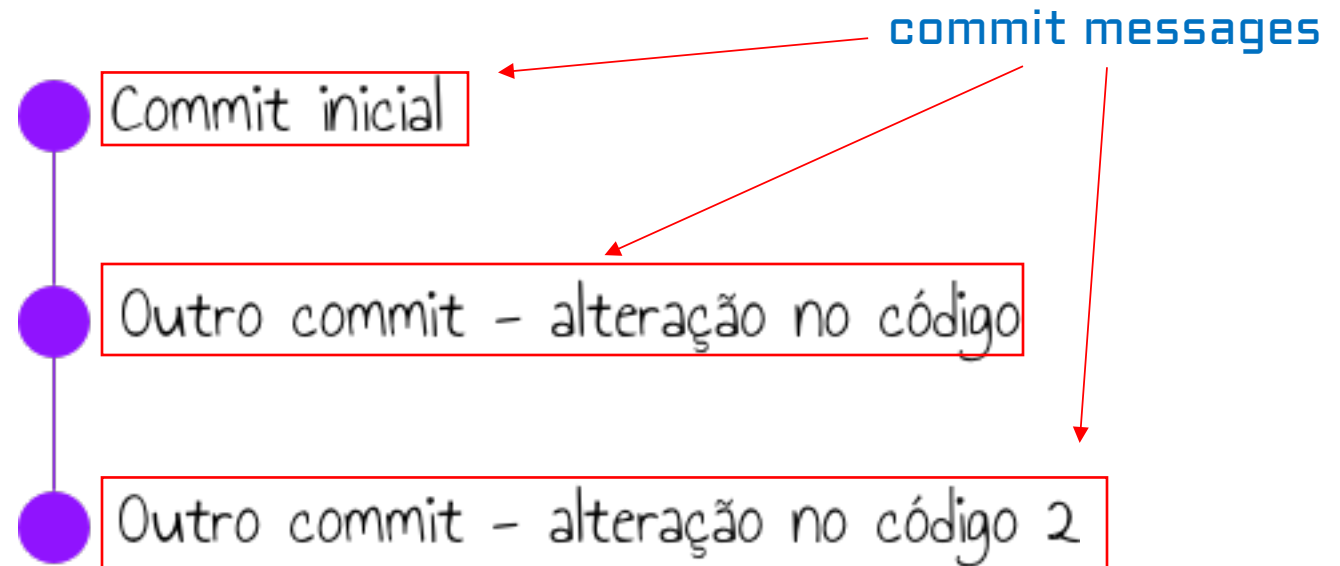
commit

Até o momento, vimos o conceito de **commit** e de **repositório**. **Commit** é o processo de salvar as alterações em um repositório. No exemplo da figura abaixo, fizemos um **commit** para adicionarmos os arquivos iniciais e em seguida mais dois **commits** com alguma alteração qualquer, criando assim uma *linha do tempo*.



commit

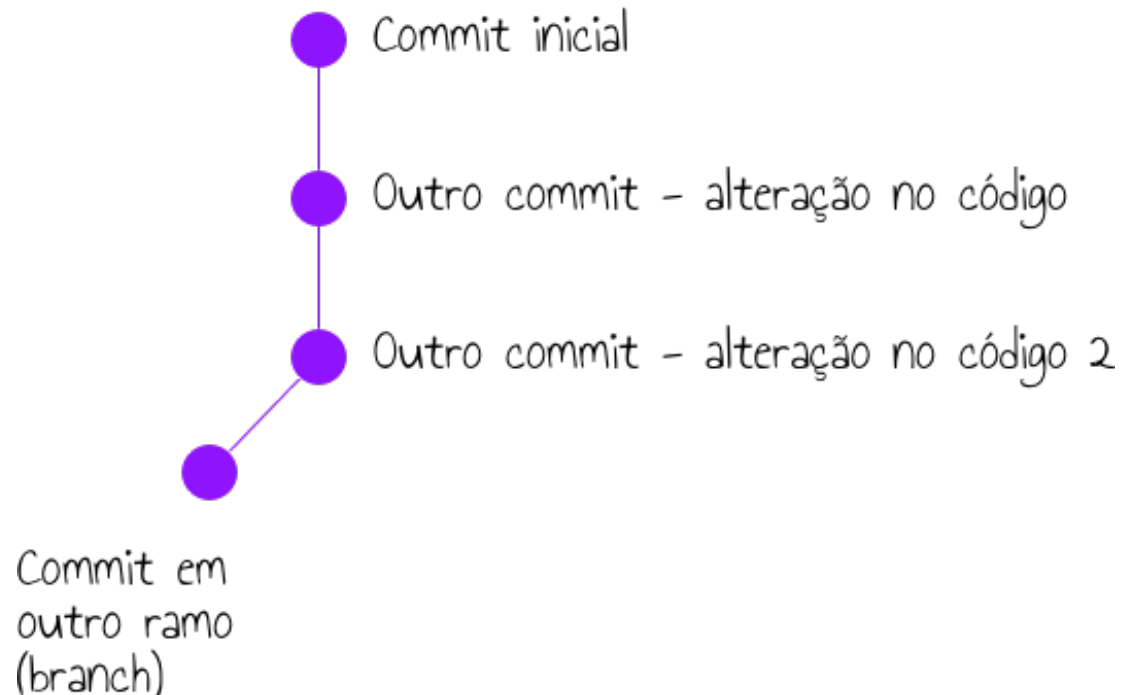
Podemos clicar em qualquer ponto na *linha do tempo* onde houve algum commit para conferir o que foi mudado ou para iniciarmos uma discussão.



Branch

Branch

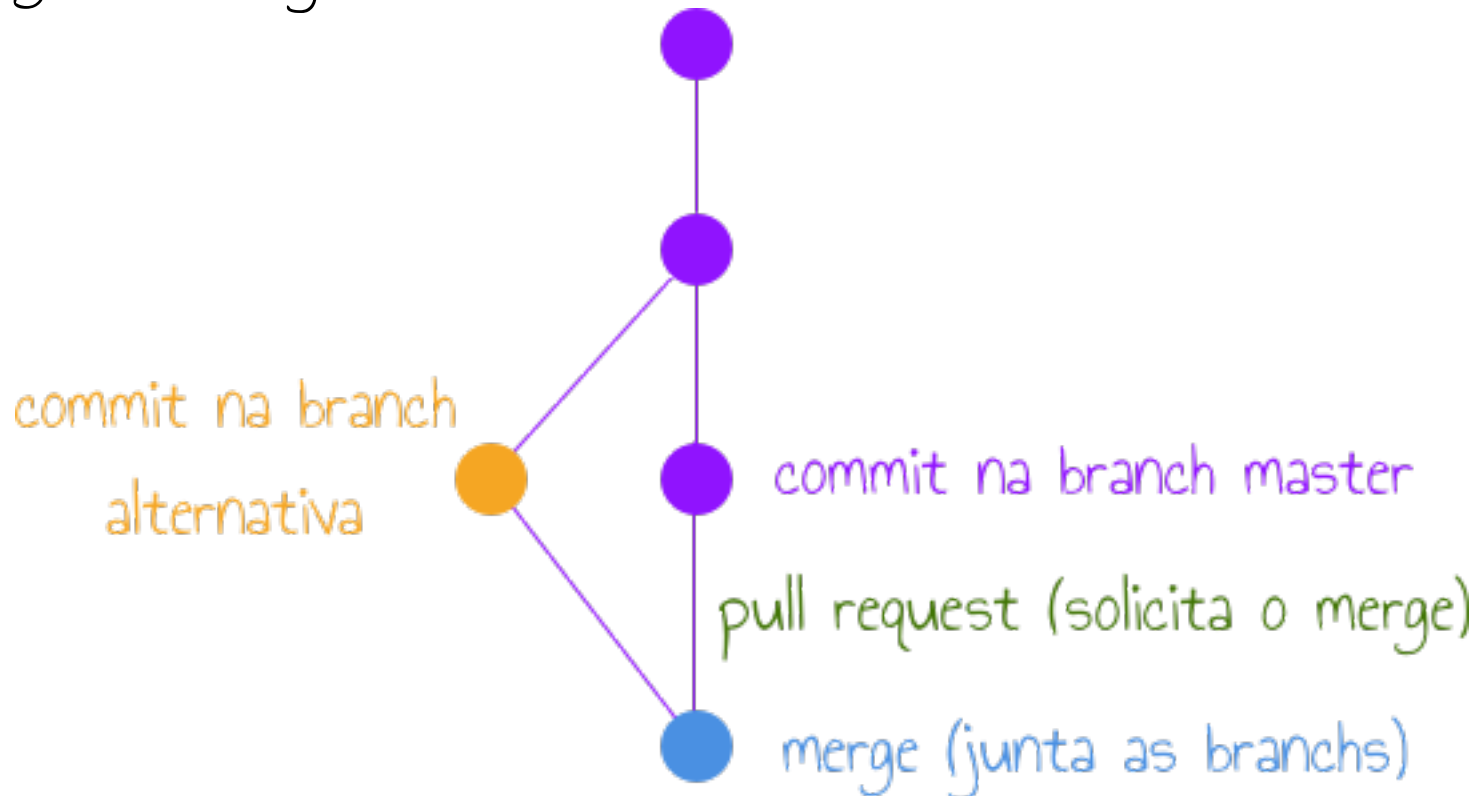
Muitas vezes, precisamos fazer testes mas sem modificar o código que já está rodando. Para isso criamos branches. Os branches são cópias do código atual. Alterações em uma branch **não interferem** na linha do tempo principal (branch master).



pull request

pull request

pull request é uma solicitação enviada ao git para unir duas branches. Quando o pull request é aceito, dizemos que os códigos foram juntados (**merge**).



pull request

É recomendável sempre fazer **pull requests** a cada pequena mudança.

outros conceitos

Outros conceitos

Na primeira parte da aula, fizemos nossos testes pelo próprio site do github.

Mas quando fazemos alterações no computador, existem algumas etapas extras que devem ser feitas e conceitos adicionais que devem ser conhecidos.

add

add

O git não verifica mudanças quando criamos novos arquivos ou pastas. É preciso informar o git quais arquivos fazem parte do repositório. Isso é chamado de **add**.

push

push

Quando comitamos alterações no site do github, as alterações automaticamente já são mostradas na página. No entanto, quando trabalhamos no computador, usamos o comando **push** para enviarmos o(s) arquivo(s) comitados para o servidor.

hash

hash

Um código de identificação é criado sempre que um commit é realizado. Este código é chamado de hash.

O hash é um código longo, composto por letras e números. Este código será útil mais tarde para identificarmos os commits. Ex:

d6cd1e2bd19e03a81132a23b2025920577f84e37

clone

clone

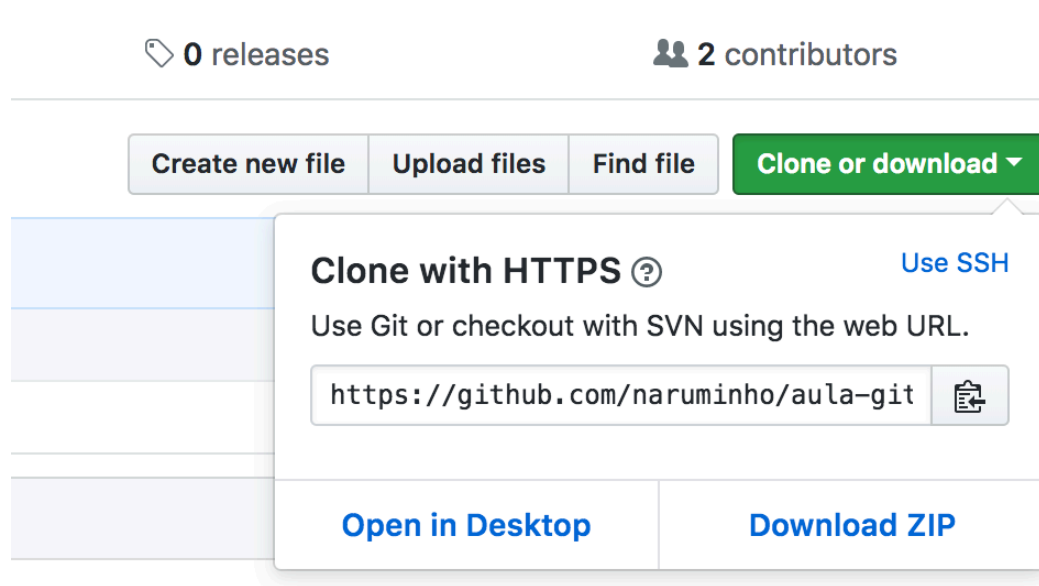
Quando fazemos um branch, criamos um novo ramo em nosso repositório. No caso do clone, criamos uma outra cópia inteira do projeto.

Podemos clonar projetos de outras pessoas e outras pessoas podem clonar nossos projetos e até mesmo sugerir modificações de modo muito semelhante ao branch. O dono do projeto (repositório), precisa aceitar as alterações.

De um ponto de vista superior, branch e clone criam espaços de códigos separados e protegidos do código principal. Isso é conhecido como **fork**.

clone

É comum clonarmos um repositório para usarmos. Ou, em muitos casos, basta fazer o download como uma pasta compactada. Tipicamente, cada membro do tipo trabalha com seu próprio repositório e merge com outros repositórios quando é necessário.



README.md

README.md

O README.md é um arquivo opcional que deve ser colocado na pasta principal do repositório. Este arquivo é lido e exibido automaticamente ao abrirmos o site do repositório. Portanto, podemos colocar nele, informações úteis sobre o projeto. Ex: descrição do projeto, instruções sobre instalação e uso.

A extensão `.md` significa que o arquivo é do tipo `markdown`. É um arquivo texto com instruções muito simples que podem ser usadas para criar um documento formatado.