



Publicar aplicações web

Prof. Renan Ponick

Roteiro das aulas / Calendário

1ª Segunda:

- Apresentação;
- Navegadores e buscadores;

2ª Segunda:

- Hospedagem, registro, tráfego;
- Pesquisa 1;

3ª Segunda:

- Aplicativos.
- Pesquisa 2;

4ª Segunda:

- Boas práticas usando site do Google SEO e W3C;

5ª Segunda:

- Git;

6ª Segunda:

- GitHub;

7ª Segunda:

- Build e Deploy;
- Backups;

8ª Segunda:

- Revisão;
- Kahoot;

9ª Segunda:

- Git Hub Pages;

10ª Segunda:

- Railway

11ª Segunda:

- Testes.

12ª Segunda:

- Avaliação Final;



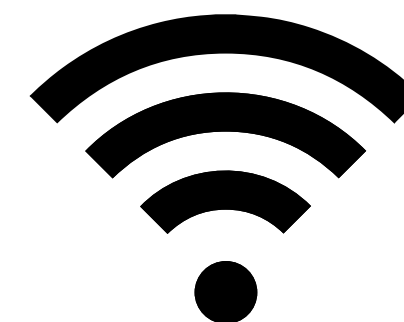
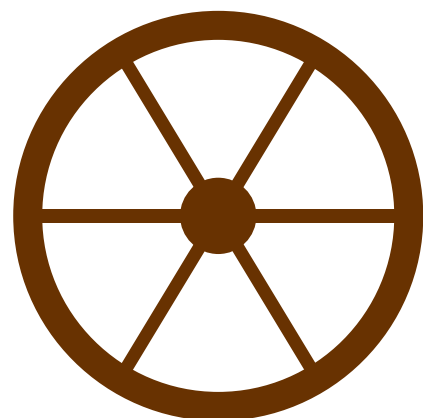
Lema

"Um desistente nunca vence, e um vencedor nunca desiste"

Napoleon Hill

O começo

Boa parte das coisas que nós conhecemos hoje surgiram de algum problema.



E o Git?

Os Problemas

- Preciso voltar o arquivo na versão que estava antes - rollback - Ctrl+Z;
- Deleção de arquivo sem a possibilidade de recuperação;
- Cópias e mais copias de um arquivos:
 - tcc.docx;
 - tcc-final.docx;
 - tcc-agora-vai.docx;



A solução

Controle de Versão

Os sistemas de controle de versão são ferramentas de software que ajudam as equipes de software a gerenciar as alterações ao código-fonte ao longo do tempo;



BitKeeper

Ferramenta de versionamento que começou a exigir pagamento das outras empresas de software para que fosse usada...



O nascimento

Descontente com as cobranças do BitKeeper, o dono do Linux criou o Git.

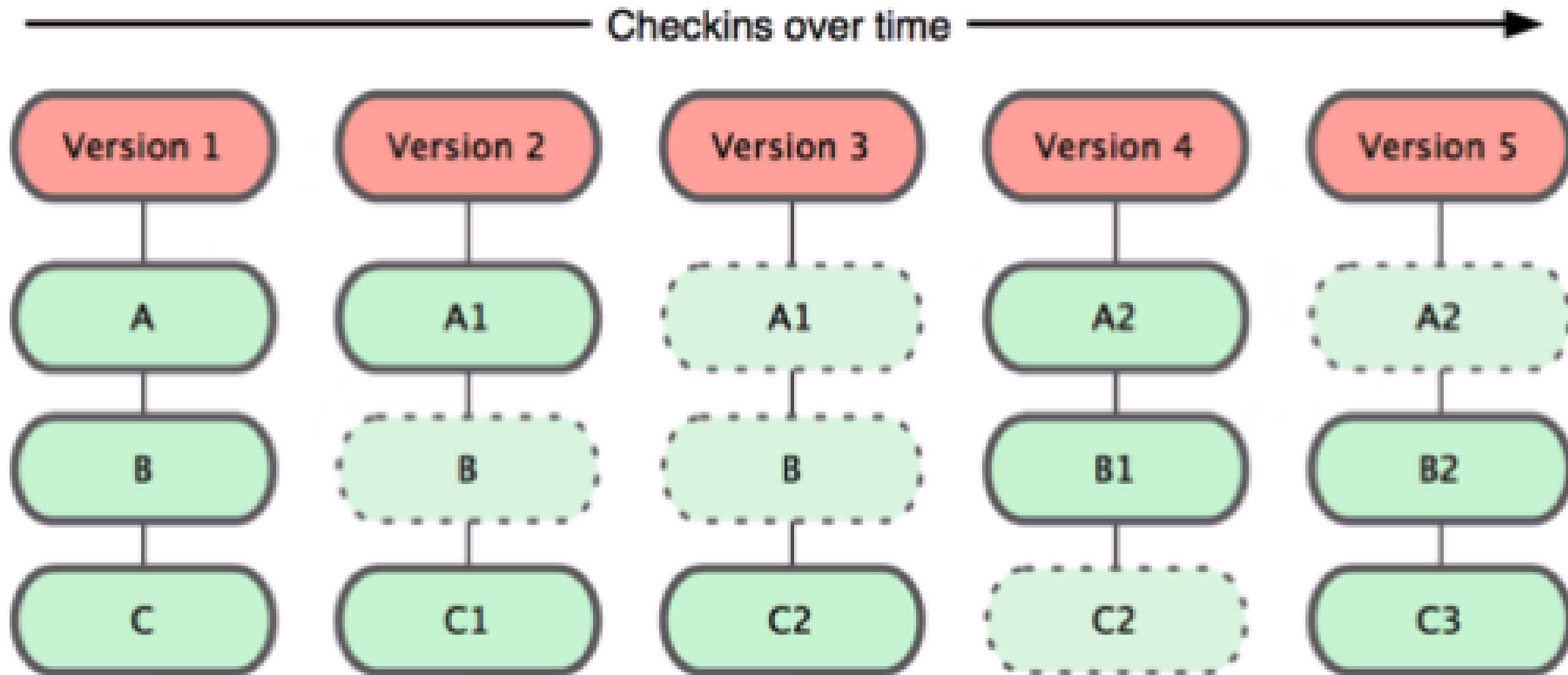
Corrigindo várias deficiências do BitKeeper.



Git

O Git ferramenta de versionamento local, que salva o estado de cada arquivo na hora do versionamento e caso o arquivo não sofra a alteração ele cria um link simbólico para o arquivo não editado.

Git



E o GitHub?

Os Problemas

- Como trabalhar em equipe, utilizando o controle do Git, para que tenhamos um controle de versão?
- Como compartilhar o conhecimento(o código fonte)?



GitHub

Serviço web compartilhado para projetos que utilizam o Git para versionamento. É um local de armazenamento de código.



git

Controle de Versão **Local**

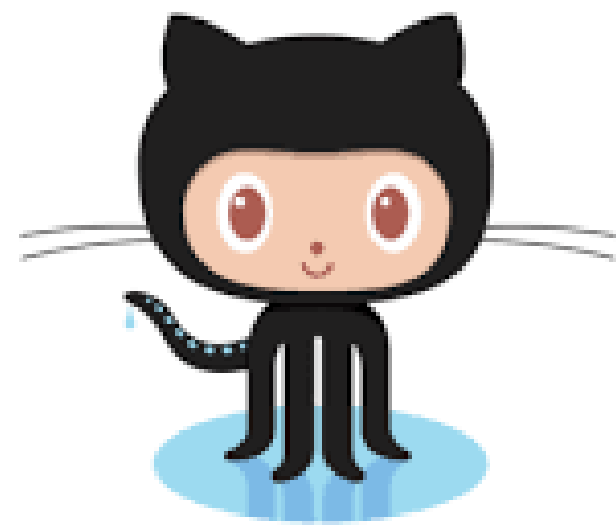
!=



GitHub

Repositório **remoto**

Check Point



GitHub



git

Quem veio primeiro?
Um poderia existir sem o outro?
Eles são os únicos?

Dia a dia



Para que, como usá-los?



Dia a Dia

O desenvolvimento de software possui uma característica forte quanto ao desenvolvimento paralelo(em conjunto) do software pela equipe de desenvolvedores.

EX: Login, Cadastro, BD



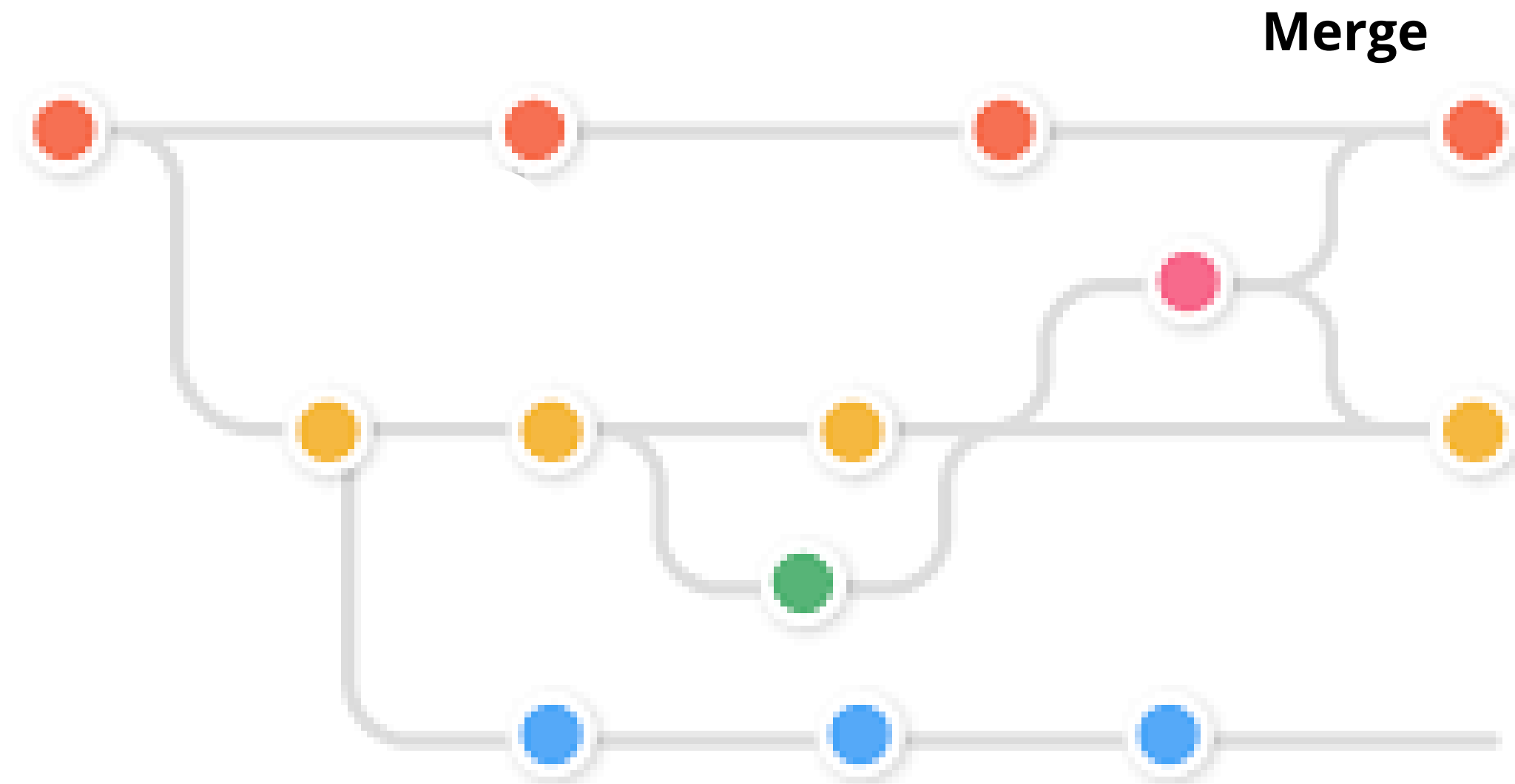
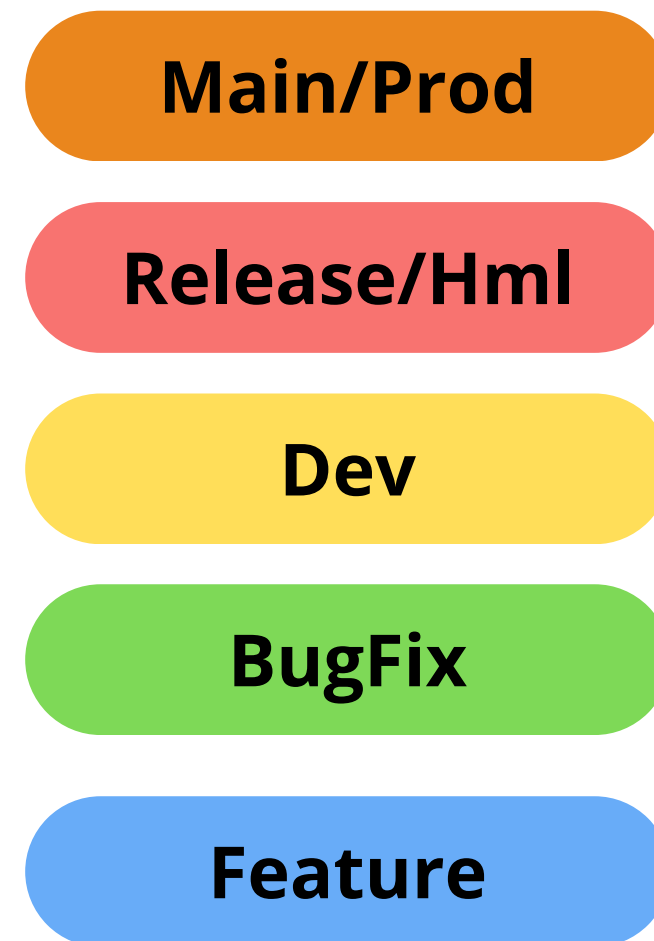


Branchs

Pesquisa rápida no google

É útil em situações nas quais você deseja adicionar um novo recurso ou corrigir um erro, gerando uma nova ramificação garantindo que o código instável não seja mesclado nos arquivos do projeto principal.

Equipe



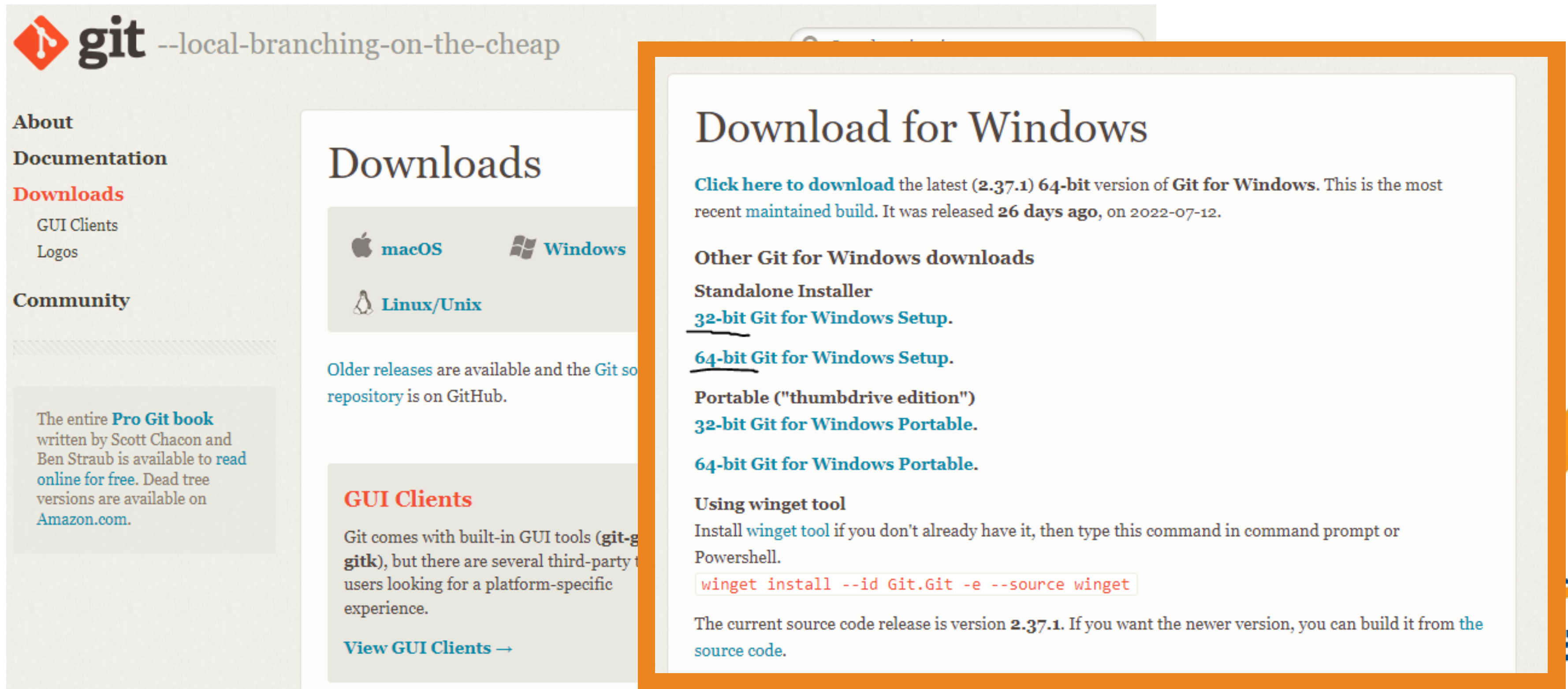


Download - Instalação Git

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

Download - Instalação

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



The image is a screenshot of the Git website's Downloads page, specifically the Windows section. The page has a light beige background with a blue and orange geometric design in the top left corner. The left sidebar contains navigation links: 'About', 'Documentation', 'Downloads' (highlighted in red), 'GUI Clients', 'Logos', and 'Community'. Below these is a box about the 'Pro Git book'. The main content area is titled 'Downloads' and features icons for macOS, Windows, and Linux/Unix. A section titled 'Older releases' mentions that older versions are available on GitHub. Below that is a 'GUI Clients' section. The right side of the page, highlighted with an orange border, is titled 'Download for Windows' and provides instructions for downloading the latest 64-bit version (2.37.1) as of July 12, 2022. It lists other download options: 'Standalone Installer' (32-bit and 64-bit Setup), 'Portable ("thumbdrive edition")' (32-bit and 64-bit Portable), and 'Using winget tool'. The winget command is shown in a code block: `winget install --id Git.Git -e --source winget`. It also mentions the current source code release is version 2.37.1.

git --local-branching-on-the-cheap

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.

Downloads

macOS Windows Linux/Unix

Older releases are available and the Git source repository is on GitHub.

GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Download for Windows

[Click here to download](#) the latest (**2.37.1**) **64-bit** version of **Git for Windows**. This is the most recent **maintained build**. It was released **26 days ago**, on 2022-07-12.

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 **2.37.1**. If you want the newer version, you can build it from [the source code](#).

Instalação

Git 2.37.1 Setup

Information

Please read the following important information before continuing.

When you are ready to continue with the installation, click Next.

GNU General Public License

Version 3, 2007

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute copies of this license document, but changing it is not allowed.

Program

The licenses for most software are not public domain. The licenses for most software are not public domain. The licenses for most software are not public domain. The licenses for most software are not public domain.

<https://gitforwindows.org/>

Select Destination Location

Where should Git be installed?



Setup will install Git in

To continue, click Next. If you

C:\Program Files\Git

At least 265,9 MB of free disk

<https://gitforwindows.org/>

Git 2.37.1 Setup

Select Components

Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

- ☐ Additional icons
 - ☐ On the Desktop
- ☒ Windows Explorer integration
 - ☒ Git Bash Here
 - ☒ Git GUI Here
- ☒ Git LFS (Large File Support)
- ☒ Associate .git* configuration files with Git
- ☒ Associate .sh files to be run with Bash
- ☐ Check daily for Git for Windows updates
- ☐ (NEW!) Add a Git Bash Profile to Windows

Current selection requires at least 265,9 MB of free disk

<https://gitforwindows.org/>

Git 2.37.1 Setup

Select Start Menu Folder

Where should Setup place the program's shortcuts?

Setup will create the program's shortcuts in the following Start Menu folder.

To continue, click Next. If you would like to select a different folder, click Browse.

Git

Browse...

☐ Don't create a Start Menu folder

<https://gitforwindows.org/>

Back

Next

Cancel

Instalação

Git 2.37.1 Setup

Choosing the default editor used by Git

Which editor would you like Git to use?

Use Vim (the ubiquitous text editor)

The [Vim editor](#), while powerful, is unintuitive and its key bindings

Note: Vim is the default editor and it is highly recommended to switch

Note: This will leave the 'core.editor' to the 'EDITOR' environment variable, which may set it to some other editor

<https://gitforwindows.org/>

Git 2.37.1 Setup

Adjusting the name of the initial branch in new repositories

What would you like Git to name the initial branch after "git init"?

☒ Let Git decide

Let Git use its default branch name in newly created repositories. The default is "main", which is a more inclusive name in the near future.

☐ Override the default branch name

NEW! Many teams already renamed "main", "trunk" and "development". What would you like the initial branch to be:

This setting does not affect existing repositories.

<https://gitforwindows.org/>

Git 2.37.1 Setup

Adjusting your PATH environment

How would you like to use Git from the command line?

☐ Use Git from Git Bash only

This is the most cautious choice as you will only be able to use the Git command from Git Bash.

☒ Git from the command line and in new consoles

(Recommended) This option adds the Git command to the PATH to avoid cluttering your environment. You will be able to use Git from Git Bash, PowerShell as well as any third-party terminal.

☐ Use Git and optional Unix tools from the Windows Command Prompt

Both Git and the optional Unix tools from MinGW will be available in the Windows Command Prompt. **Warning: This will override Windows PATH, which could result in unwanted behavior. Use this option if you understand the risks.**

<https://gitforwindows.org/>

Git 2.37.1 Setup

Choosing the SSH executable

Which Secure Shell client program would you like Git to use?

☒ Use bundled OpenSSH

This uses ssh.exe that comes with Git.

☐ Use (Tortoise)Plink

To use PuTTY, specify the path to an existing copy of (Tortoise)Plink.exe:

...

☐ Set ssh.variant for Tortoise Plink

☐ Use external OpenSSH

NEW! This uses an external ssh.exe. Git will not install its own OpenSSH (and related) binaries but use them as found on the PATH.

<https://gitforwindows.org/>

Back

Next

Cancel

Instalação

The image displays four overlapping windows from the Git 2.37.1 Setup wizard, each showing a different configuration step. The windows are titled 'Git 2.37.1 Setup' and feature the Git logo in the top right corner. The first window, 'Choosing HTTPS transport backend', asks 'Which SSL/TLS library would you like Git to use for HTTPS connections?' and has two radio button options: 'Use the OpenSSL library' (selected) and 'Use the native Windows SChannel library'. The second window, 'Configuring the line ending conversions', asks 'How should Git treat line endings in text files?' and has three radio button options: 'Checkout Windows-style' (selected), 'Checkout as-is, commit LF', and 'Checkout as-is, commit CRLF'. The third window, 'Configuring the terminal emulator to use with Git Bash', asks 'Which terminal emulator do you want to use with your Git Bash?' and has two radio button options: 'Use MinTTY (the default terminal emulator)' (selected) and 'Use Windows' default console window'. The fourth window, 'Choose the default behavior of `git pull`', asks 'What should `git pull` do by default?' and has three radio button options: 'Default (fast-forward or merge)' (selected), 'Rebase', and 'Only ever fast-forward'. Each window also includes a URL 'https://gitforwindows.org/' at the bottom. The fourth window has 'Back', 'Next', and 'Cancel' buttons at the bottom right.

Git 2.37.1 Setup

Choosing HTTPS transport backend
Which SSL/TLS library would you like Git to use for HTTPS connections?

☒ **Use the OpenSSL library**
Server certificates will be validated by the OpenSSL library.

☐ **Use the native Windows SChannel library**
Server certificates will be validated by the native Windows SChannel library. This option also allows you to use certificates distributed e.g. via Active Directory.

<https://gitforwindows.org/>

Git 2.37.1 Setup

Configuring the line ending conversions
How should Git treat line endings in text files?

☒ **Checkout Windows-style**
Git will convert LF to CRLF in text files, CRLF will be converted to LF in this is the recommended setting.

☐ **Checkout as-is, commit LF**
Git will not perform any conversion when committing text files, CRLF in text files will be converted to LF this is the recommended setting.

☐ **Checkout as-is, commit CRLF**
Git will not perform any conversion when committing text files. Choosing this option is only useful for projects ("core.autocrlf" is set to false).

<https://gitforwindows.org/>

Git 2.37.1 Setup

Configuring the terminal emulator to use with Git Bash
Which terminal emulator do you want to use with your Git Bash?

☒ **Use MinTTY (the default terminal emulator)**
Git Bash will use MinTTY as terminal emulator. Non-rectangular selections and a Unicode cursor (as interactive Python) must be launched.

☐ **Use Windows' default console window**
Git will use the default console window with Win32 console programs such as very limited default scroll-back, needs order to display non-ASCII characters. The default console window was not freely resizable and it is not recommended.

<https://gitforwindows.org/>

Git 2.37.1 Setup

Choose the default behavior of `git pull`
What should `git pull` do by default?

☒ **Default (fast-forward or merge)**
This is the standard behavior of `git pull`: fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.

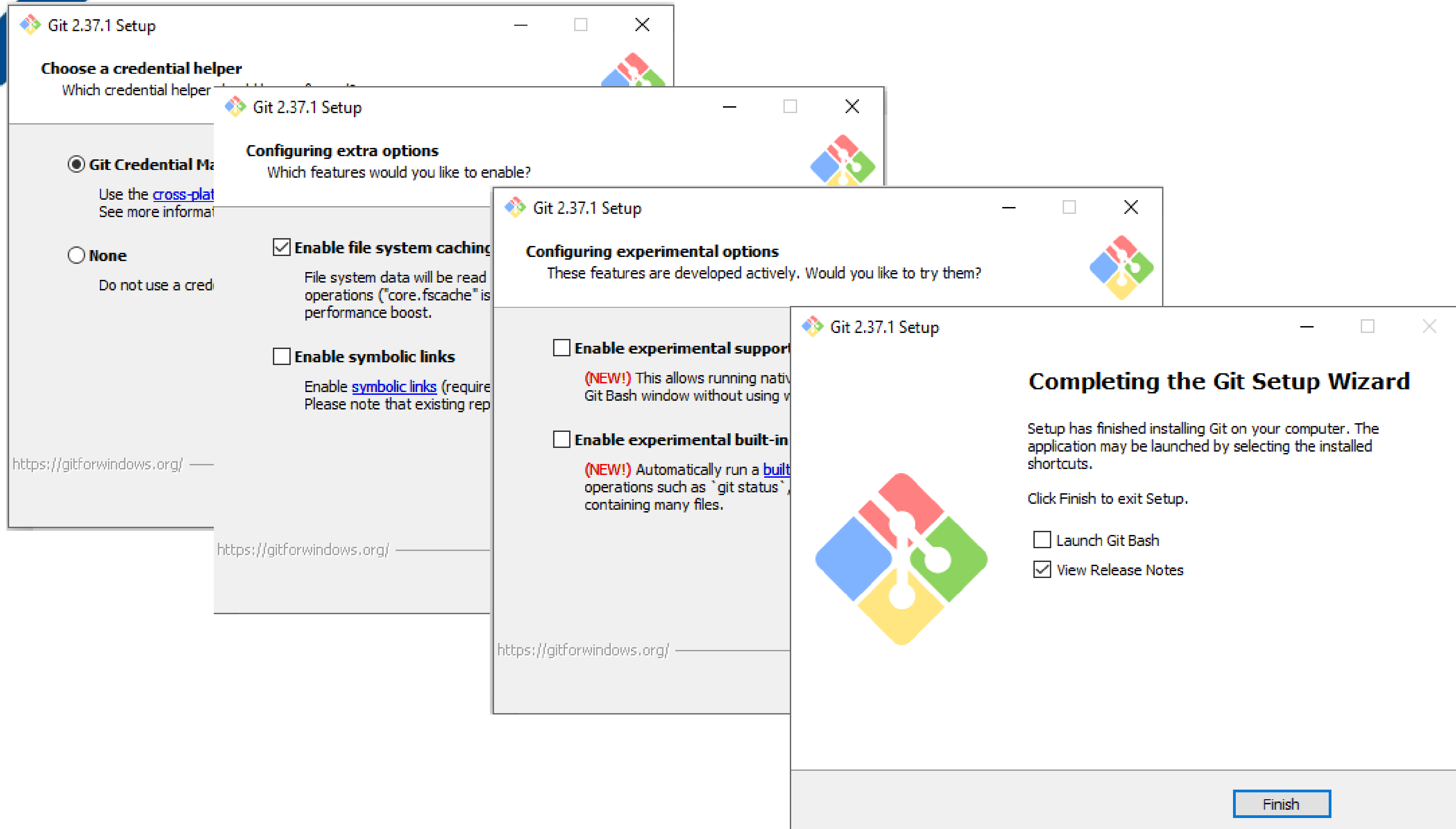
☐ **Rebase**
Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.

☐ **Only ever fast-forward**
Fast-forward to the fetched branch. Fail if that is not possible.

<https://gitforwindows.org/>

Back Next Cancel

Instalação



Verificação

Abra o CMD, Terminal de Comando, Shell ou PowerShell
Digite git --version

```
C:\Users\Renan Ponick>git --version  
git version 2.37.1.windows.1
```

Configuração

```
git config --global user.name "Seu Nome"  
git config --global user.email "seuEmail@gmail.com"
```

Inicializando - Local

```
C:\Users\Renan Ponick>cd Documents  
C:\Users\Renan Ponick\Documents>mkdir exemploGit  
C:\Users\Renan Ponick\Documents>cd exemploGit  
C:\Users\Renan Ponick\Documents\exemploGit>git init  
Initialized empty Git repository in C:/Users/Renan Ponick/Documents/exemploGit/.git/
```



Crie uma conta no GitHub

<https://github.com/>

Mão na massa

Pesquise sobre os seguintes comandos:

`git status`

`git add <filename ou . >`

`git restore --staged <filename ou . >`

`git branch <branchname>`

`git checkout <branchname>`

`git checkout -b <branchname>`

`git commit -m "<description>"`

`git push`

`git branch -D <branchname>`

`git fetch`

`git pull`