

# Programação Aplicada Ao Direito



U F *m* G



**MARCELO CANDIOTTO**  
Sociedade de Advogados **MCSA**



ASSOCIAÇÃO  
BRASILEIRA DE  
**LAWTECHS &  
LEGALTECHS**



# Conteúdo

## Dia 09 - Versionamento e repositório de Código

- Controle descentralizado – GIT
- Repositórios WEB – Github
- Controle de casos e wiki
- Exercícios Práticos: Criação de sua conta no Github e seu primeiro commit

# Controle descentralizado - GIT

## O que é o GIT?

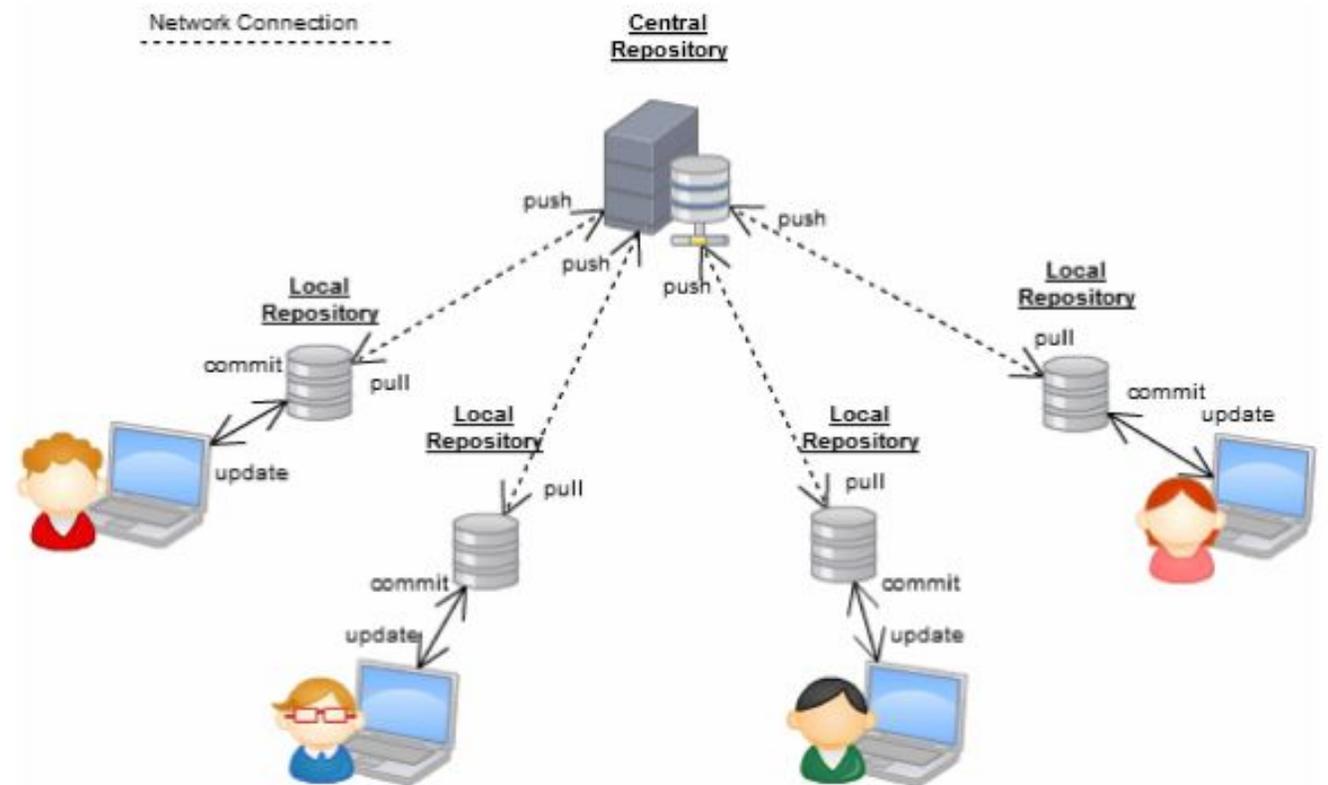
- É um sistema open-source que faz o controle de versão de arquivos.
- Por meio dele podemos desenvolver softwares na qual diversos programadores podem contribuir simultaneamente no mesmo projeto, editando e criando novos arquivos e permitindo que os mesmos possam existir sem o risco de suas alterações serem sobrescritas.
- O Git foi inicialmente projetado e desenvolvido por **Linus Torvalds** para o desenvolvimento do **kernel Linux**.



# Controle descentralizado - GIT

## Controle de Versões

- Sistema que registra as alterações.
- Permite o desenvolvimento colaborativo.
- Permite que você saiba quem fez as alterações e quando.
- Permite reverter as alterações e voltar ao estado anterior.



# Instalando o GIT

## Links de instalação:

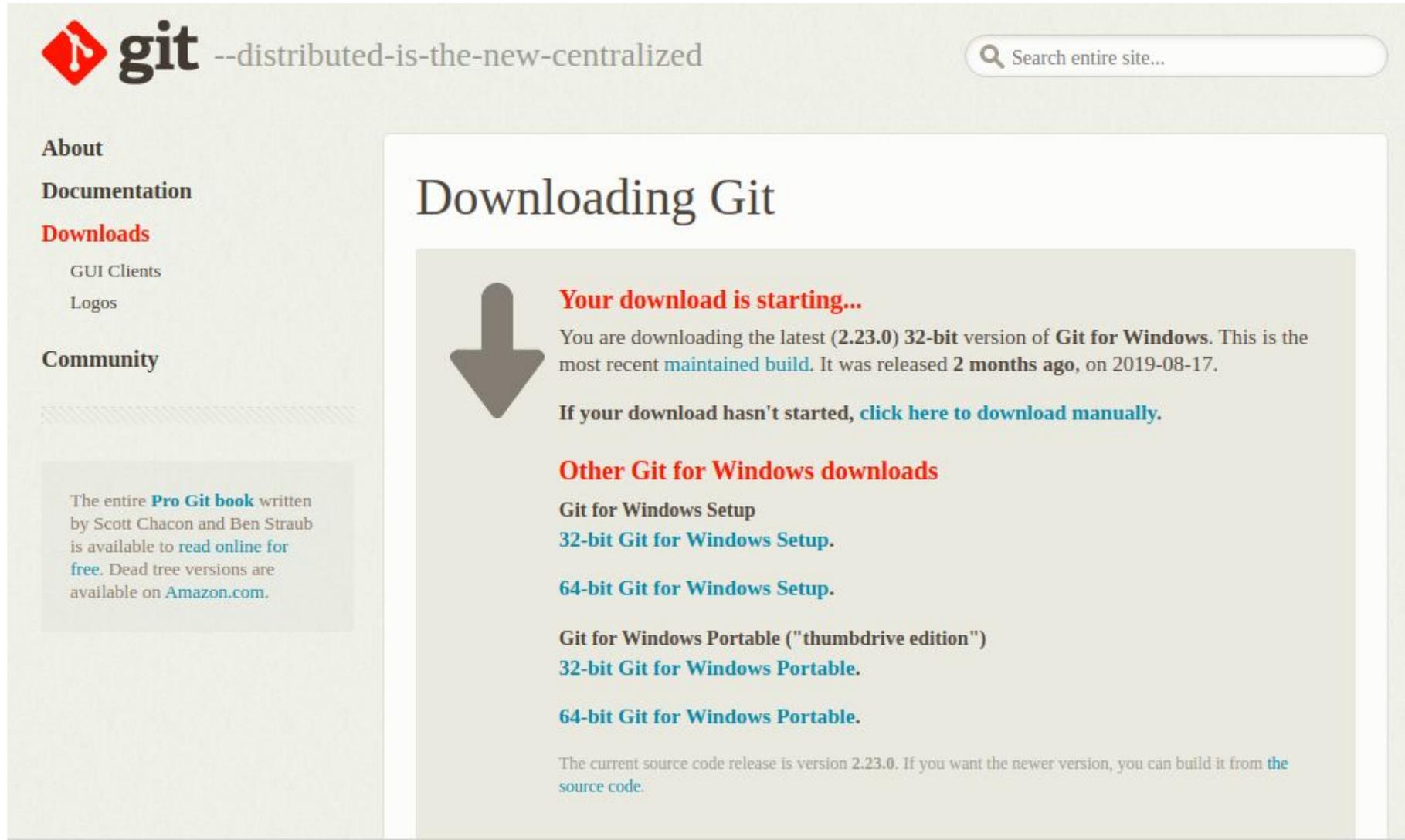
- **Linux (Debian)**
  - Digitar comando no terminal: *sudo apt-get install git*
- **Linux (Fedrora)**
  - Digitar comando no terminal: *sudo yum install git*
- **Mac**
  - <https://git-scm.com/download/mac>
- **Windows**
  - <https://git-scm.com/download/win>

# Instalando o GIT

The screenshot shows the Git website homepage. At the top left is the Git logo (a red diamond with a white branching symbol) followed by the text "git --everything-is-local". To the right is a search bar with the placeholder text "Search entire site...". Below the logo are navigation links: "About", "Documentation", "Downloads" (highlighted in red), "GUI Clients", and "Logos". A "Community" section is also visible. The main content area features a large "Downloads" heading. Below it, there are three platform-specific download options: "Mac OS X" (with an Apple logo), "Windows" (with a Windows logo), and "Linux/Unix" (with a Tux penguin logo). To the right of these options is a graphic of a computer monitor displaying "Latest source Release 2.23.0" and "Release Notes (2019-08-16)", with a prominent button that says "Downloads for Linux". Below the download options, a text block states: "Older releases are available and the Git source repository is on GitHub." At the bottom of the page, there are two columns: "GUI Clients" and "Logos", each with a brief description and a "View" link with a right-pointing arrow.



# Instalando o Git: Windows



The screenshot shows the Git website's 'Downloading Git' page for Windows. The page features a navigation menu on the left with links for 'About', 'Documentation', 'Downloads', 'GUI Clients', 'Logos', and 'Community'. The main content area is titled 'Downloading Git' and includes a large downward-pointing arrow icon. The text indicates that the user is downloading the latest (2.23.0) 32-bit version of Git for Windows, which is the most recent maintained build, released 2 months ago on 2019-08-17. It provides a link to download manually if the automatic download hasn't started. Below this, there are links for 'Other Git for Windows downloads', including 'Git for Windows Setup', '32-bit Git for Windows Setup', '64-bit Git for Windows Setup', 'Git for Windows Portable ("thumbdrive edition")', '32-bit Git for Windows Portable', and '64-bit Git for Windows Portable'. At the bottom, it mentions that the current source code release is version 2.23.0 and provides a link to the source code.

**git** --distributed-is-the-new-centralized

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](#).

## Downloading Git

**Your download is starting...**

You are downloading the latest (2.23.0) 32-bit version of **Git for Windows**. This is the most recent [maintained build](#). It was released **2 months ago**, on 2019-08-17.

If your download hasn't started, [click here to download manually](#).

### Other Git for Windows downloads

**Git for Windows Setup**

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

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

**Git for Windows Portable ("thumbdrive edition")**

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

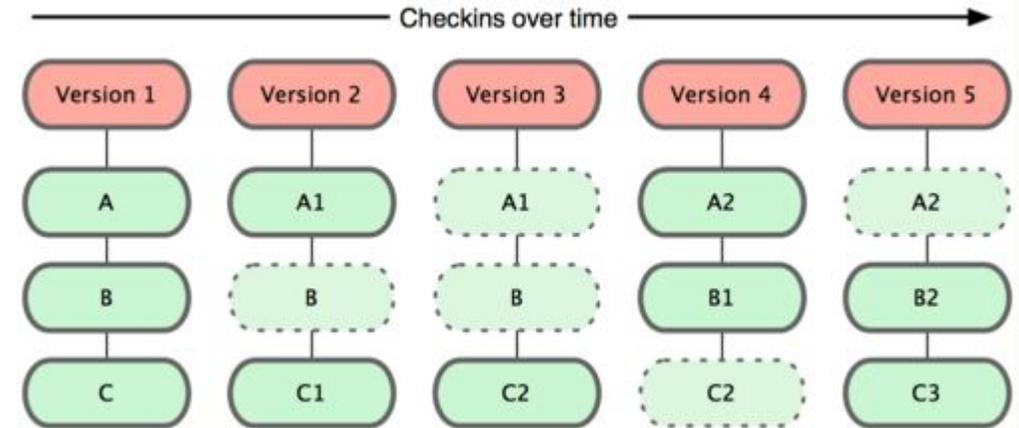
[64-bit Git for Windows Portable.](#)

The current source code release is version 2.23.0. If you want the newer version, you can build it from [the source code](#).

# Controle descentralizado - GIT

## “Snapshots”

- É a maneira como o git acompanha seu histórico de código.
- Registra essencialmente como são todos os seus arquivos em um determinado momento (de um mini-sistema de arquivos captura de algo em um determinado instante, como em uma foto).
- Para ser eficiente, se nenhum arquivo foi alterado, a informação não é armazenada novamente.



# Controle descentralizado - GIT

## Adicionar

- Para fazer o registro dos arquivos (snapshot) utilize o comando:
  - `git add nome_do_arquivo`
- Você decide quando fazer um “snapshot”.

```
Gabriellas-Air:hangman gabi$ git status
On branch styling
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

   modified:   app.rb
   modified:   lib/hangman_rules.rb
   modified:   lib/word_formatter.rb
   modified:   spec/app_spec.rb
   modified:   spec/hangman_rules_spec.rb
   modified:   spec/word_formatter_spec.rb
   modified:   views/game_over.erb
   modified:   views/games_won.erb
   modified:   views/home.erb

no changes added to commit (use "git add" and/or "git commit -a")
Gabriellas-Air:hangman gabi$
```

# Controle descentralizado - GIT

## “Commit”

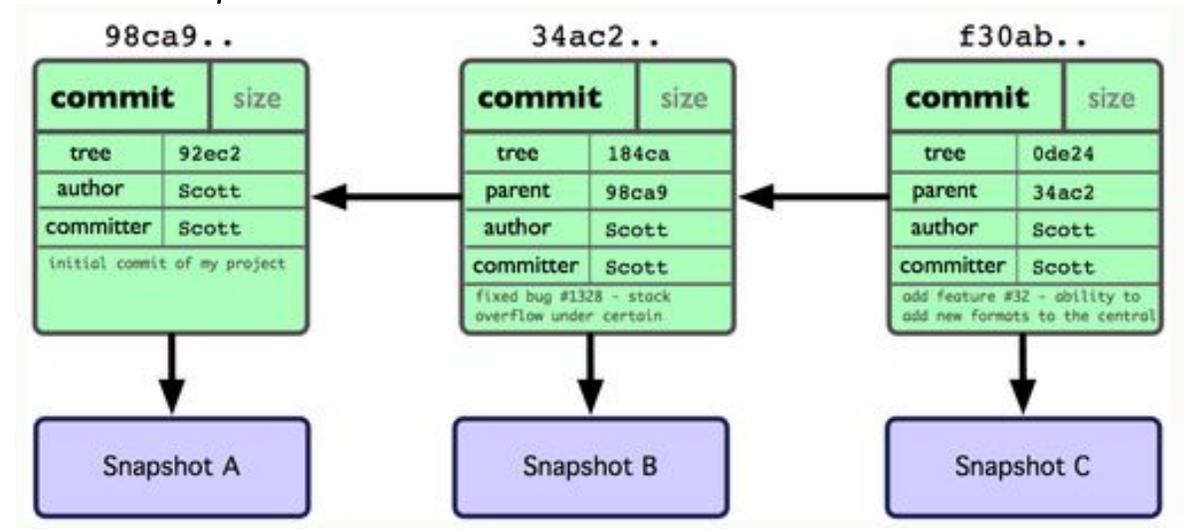
- é o ato de criar um snapshot.
- o programador submete as alterações do código fonte ao repositório e faz com que estas alterações se tornem parte da versão principal (head) do repositório.
- Essencialmente um projeto é composto por vários commits.



# Controle descentralizado - GIT

## O Commit contém 3 tipos de informações:

1. Informações sobre como os arquivos foram alterados de anteriormente.
2. Uma referência ao commit que veio antes dele, chamado de "Commit Pai".
3. Um código identificador:
  - a. Exemplo: fb2d2ec5069fc6776c8...



# Controle descentralizado - GIT

## Repositórios:

- Uma coleção de todos os arquivos e histórico desses arquivos.
  - Consiste em todos os seus commits;
  - Local onde todo o seu trabalho é armazenado;
- Pode viver em uma máquina local ou em um servidor remoto (GitHub!).
- O ato de copiar um repositório de um servidor remoto é chamado de clonagem.
- A clonagem de um servidor remoto permite que as equipes trabalhem juntas.

# Repositórios Web - Github

## O que é o Github ?

- <https://www.youtube.com/watch?v=w3jLJU7DT5E>
- Maior serviço de hospedagem de repositórios web.
- Permite a colaboração de código com qualquer pessoa on-line.
- Adiciona funcionalidade extra ao GIT:
  - Issues, Projetos, Wiki



# Repositórios Web - Github

## Criando conta no Github:

- Acessar link abaixo:
  - <https://github.com>

Why GitHub? ▾ Enterprise Explore ▾ Marketplace Pricing ▾

Search GitHub / Sign in Sign up

## Built for developers

GitHub is a development platform inspired by the way you work. From **open source** to **business**, you can host and review code, manage projects, and build software alongside 40 million developers.

Username

Email

Password

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

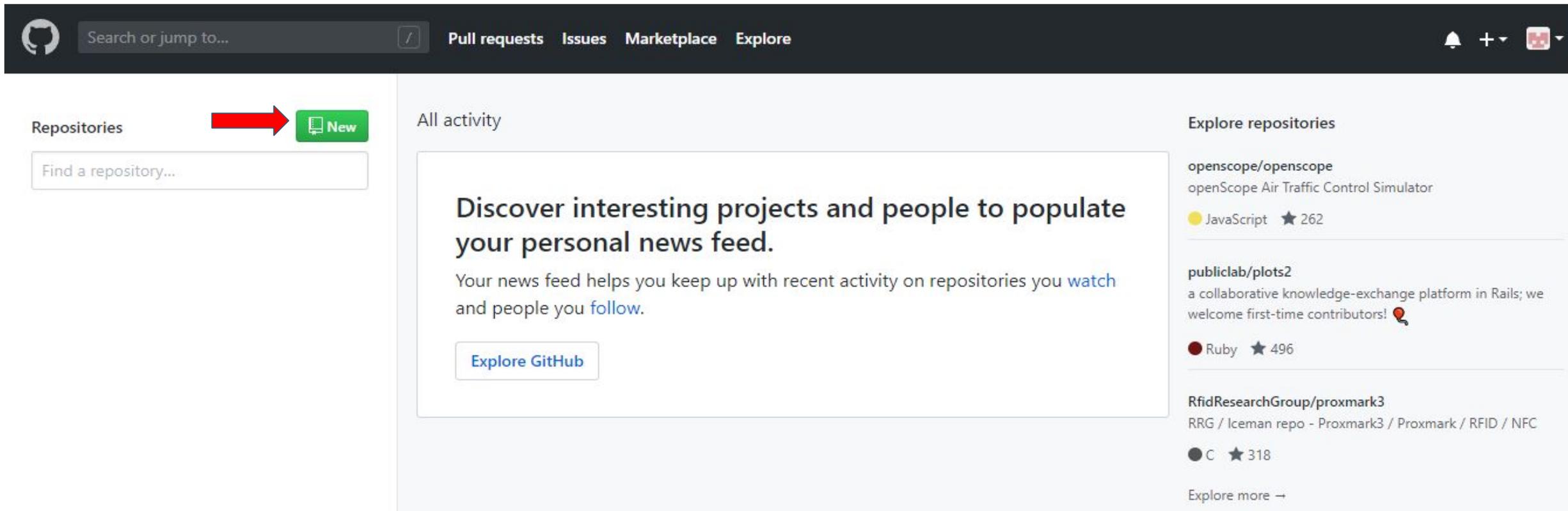
[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.

# Repositórios Web - Github

## Criando primeiro Repositório:

- Clicar em “NEW”:



The screenshot shows the GitHub homepage. At the top, there is a dark navigation bar with the GitHub logo, a search bar, and links for Pull requests, Issues, Marketplace, and Explore. Below the navigation bar, the main content area is divided into three columns. The left column is titled 'Repositories' and contains a search bar with the placeholder text 'Find a repository...'. A red arrow points to a green 'New' button with a plus icon. The middle column is titled 'All activity' and contains a large text box with the heading 'Discover interesting projects and people to populate your personal news feed.' and a button labeled 'Explore GitHub'. The right column is titled 'Explore repositories' and lists several repositories, including 'openscope/openscope' (JavaScript, 262 stars) and 'publiclab/plots2' (Ruby, 496 stars).

# Repositórios Web - Github

## Preencher Informações:

- Nome:
- Descrição (opcional)
- Público

## Arquivos Comuns:

- Readme
- Gitignore
- License

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner

Repository name \*

 programacaodireito /



Great repository names are short and memorable. Need inspiration? How about [scaling-octo-spoon?](#)

Description (optional)



Public

Anyone can see this repository. You choose who can commit.



Private

You choose who can see and commit to this repository.

Skip this step if you're importing an existing repository.

Initialize this repository with a README

This will let you immediately clone the repository to your computer.



Add .gitignore: None

Add a license: None



Create repository

# Repositórios Web - Github

## Tela Inicial do repositório:

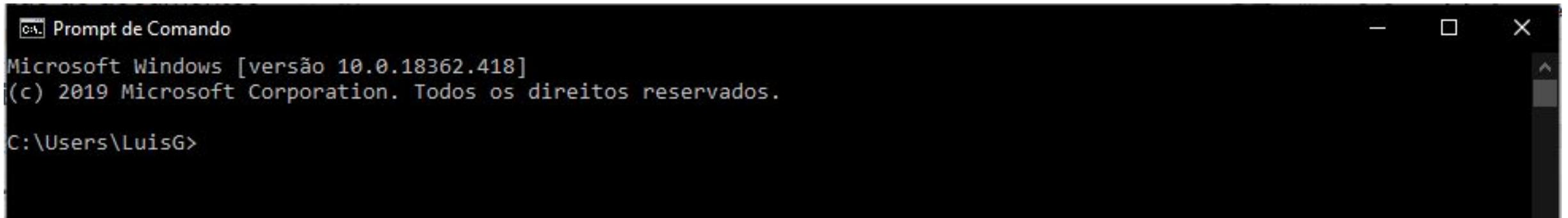
The screenshot shows the GitHub interface for a repository named 'programacaodireito / curso'. At the top, there are navigation tabs for 'Code', 'Issues 0', 'Pull requests 0', 'Projects 0', 'Wiki', 'Security', 'Insights', and 'Settings'. On the right, there are buttons for 'Unwatch 1', 'Star 0', and 'Fork 0'. Below the navigation, the repository name 'Programação Aplicada ao Direito' is displayed with an 'Edit' button. A 'Manage topics' link is also present. A summary bar shows '1 commit', '1 branch', '0 releases', and '1 contributor'. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and a prominent green 'Clone or download' button. The commit history shows an 'Initial commit' by 'programacaodireito' 6 minutes ago. Below the commit history, the 'README.md' file is shown with its content: 'curso' followed by 'Programação Aplicada ao Direito'.

# Clonando Projeto do Github

**1. Criar pasta “Curso” na área de trabalho.**

**2. Abrir terminal:**

- Digitar “cmd” em campo de pesquisa do windows e clicar em “Prompt de Comando”.

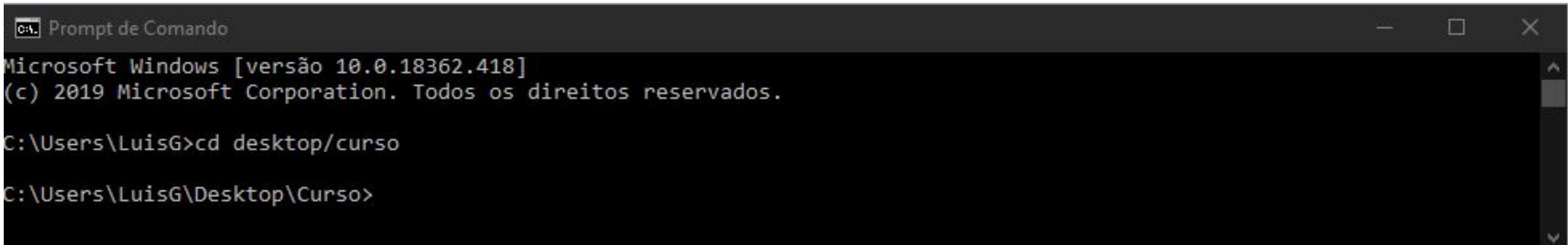
A screenshot of a Windows Command Prompt window. The title bar reads "C:\> Prompt de Comando". The main text area shows the following text: "Microsoft Windows [versão 10.0.18362.418]", "(c) 2019 Microsoft Corporation. Todos os direitos reservados.", and "C:\Users\LuisG>". The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
C:\> Prompt de Comando
Microsoft Windows [versão 10.0.18362.418]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.
C:\Users\LuisG>
```

# Clonando Projeto do Github

## 3. Acessar a pasta “Curso” pelo terminal

- Digitar o comando: `cd desktop/curso`



```
C:\> Prompt de Comando
Microsoft Windows [versão 10.0.18362.418]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.

C:\Users\LuisG>cd desktop/curso

C:\Users\LuisG\Desktop\Curso>
```

# Clonando Projeto do Github

## 4. Clonar repositório remoto ao local:

- Digitar comando: `git clone url_do_github`

```
C:\> Prompt de Comando
Microsoft Windows [versão 10.0.18362.418]
(c) 2019 Microsoft Corporation. Todos os direitos reservados.

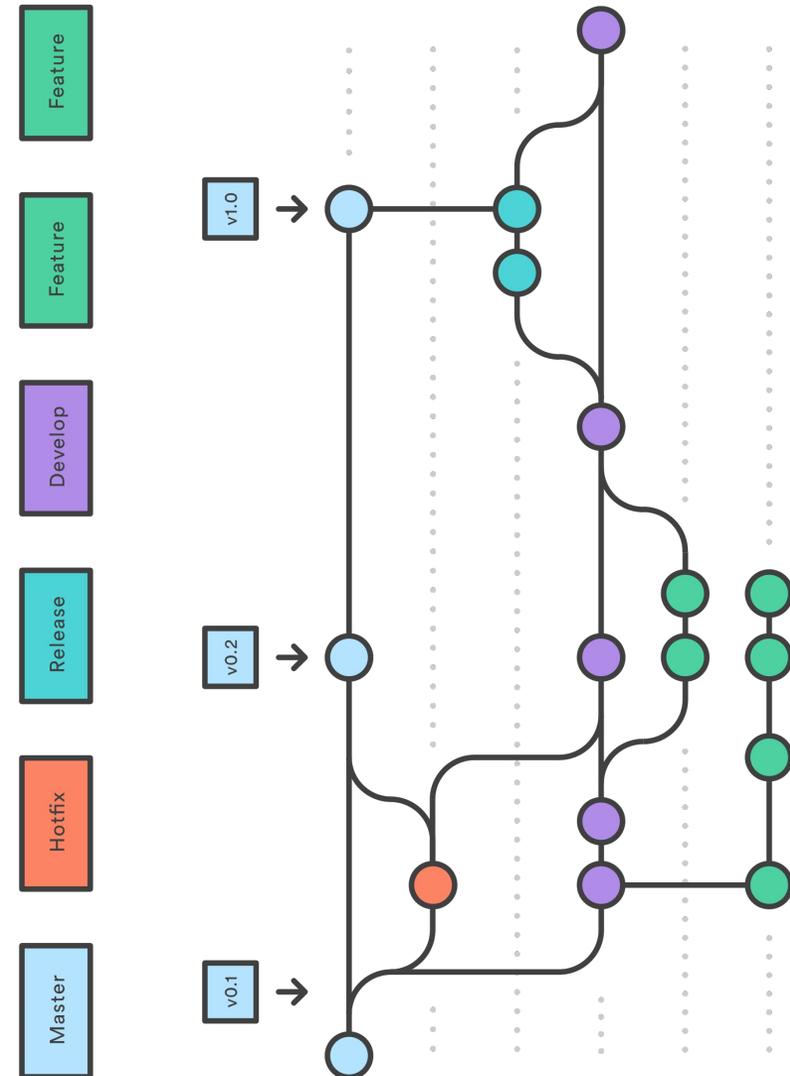
C:\Users\LuisG>cd desktop/curso

C:\Users\LuisG\Desktop\Curso>git clone https://github.com/programacaodireito/direito.git
```

# Controle descentralizado - GIT

## “Branchs” : Ramificando

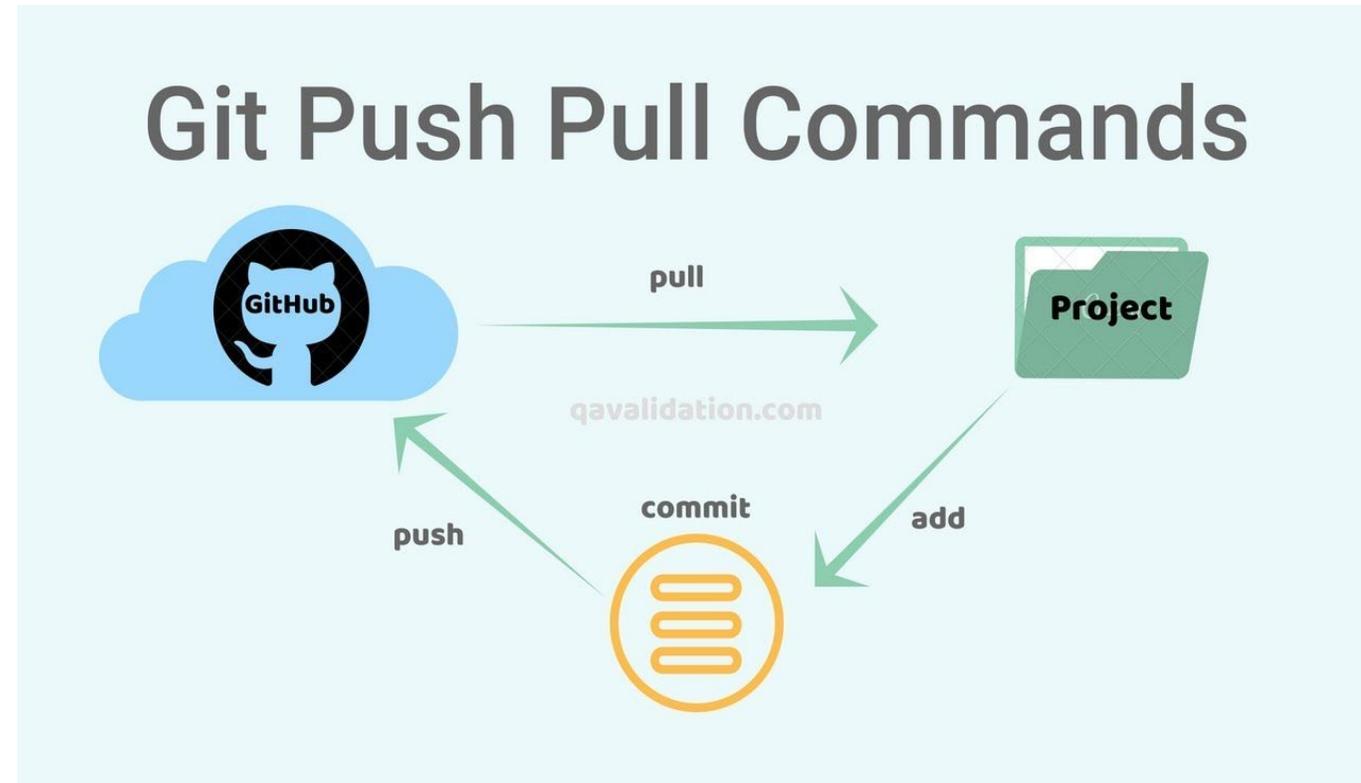
- Todos os commits no git vivem em algum ramo.
- O branch master é o branch "principal" quando você cria um repositório.
- Use outros branches para desenvolver outras funcionalidades e mescle-os (merge) ao branch master após a conclusão.



# Controle descentralizado - GIT

## Atualizando Repositórios:

- **PULL:** O processo de download de confirmações que não existem na sua máquina a partir de um repositório remoto é chamado de puxar alterações.
- **PUSH:** The process of adding your local changes to the remote repository is called pushing changes



# Issues

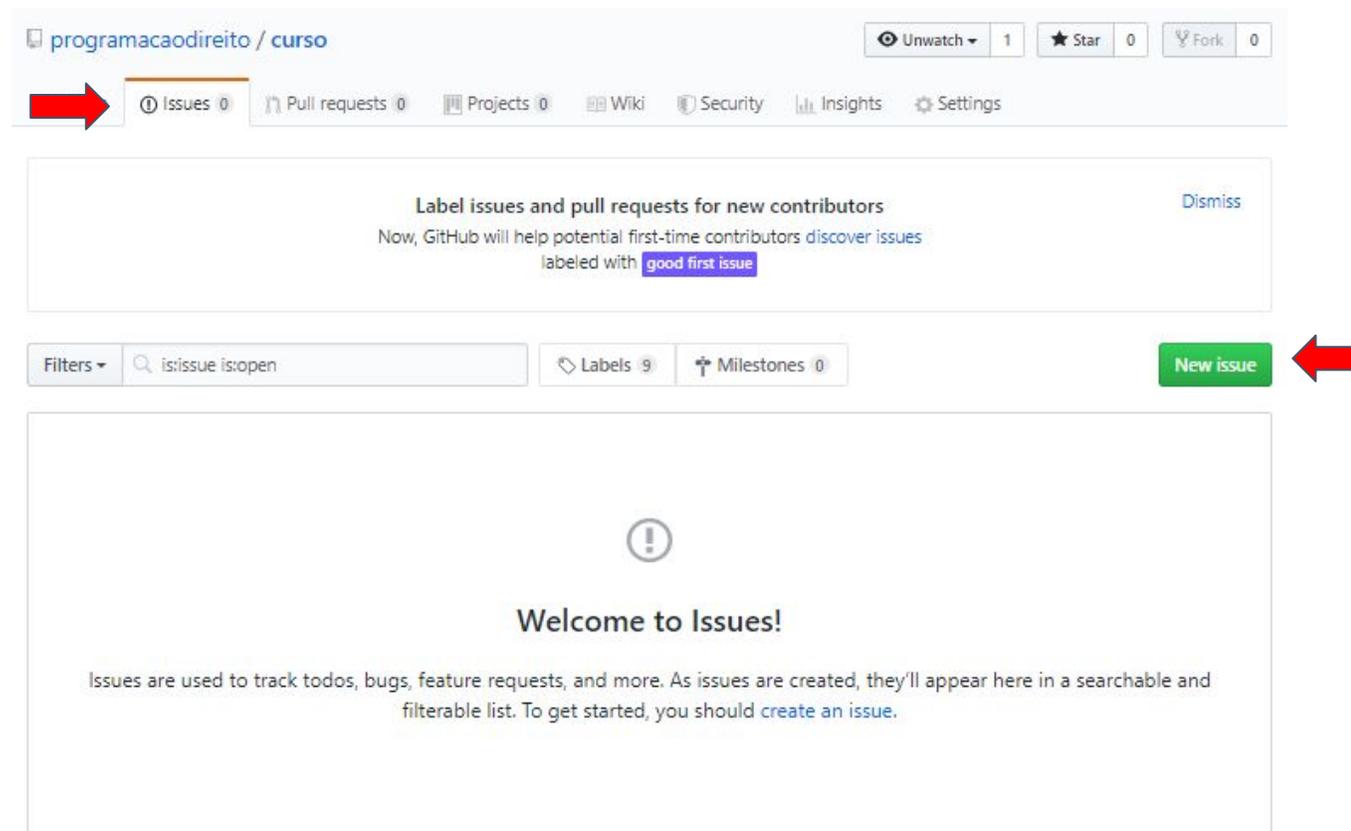
## O que é?

- As issues são os casos a serem resolvidos.
- Com elas nós descrevemos os problemas que temos nos projetos para que sejam resolvidos por qualquer pessoa que queira contribuir.
- Na metodologia scrum, fazem parte do backlog do projeto e quando priorizados, entram para o sprint.

# Issues

## Criando issue no Github:

1. Acessar a aba "issues" no repositório do Github e clicar em "New issue":



# Issues

## Criando issue no Github:

2. Definir o título e a descrição (problema, o que deverá ser criado, anexos):

The screenshot shows the GitHub interface for creating a new issue. At the top, the repository name 'programacaodireito / curso' is displayed, along with 'Unwatch' (1), 'Star' (0), and 'Fork' (0) buttons. Below this is a navigation bar with 'Code', 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', 'Security', 'Insights', and 'Settings' tabs. The main form is titled 'New Issue' and has a 'Title' input field. To the right of the title field is a 'Related issues' section with a 'Beta' badge and a 'Try it.' link. Below the title field is a 'Write' tab and a 'Preview' tab. The 'Write' tab is active, showing a rich text editor with a 'Leave a comment' placeholder. To the right of the editor is a sidebar with settings for 'Assignees' (No one—assign yourself), 'Labels' (None yet), 'Projects' (None yet), and 'Milestone' (No milestone). At the bottom of the form is a 'Submit new issue' button. Two red arrows point to the title field and the text area.

# Issues

## Criando issue no Github:

### 3. Assumindo a tarefa:

The screenshot shows the GitHub interface for creating a new issue. At the top, the repository name 'programacaodireito / curso' is displayed, along with 'Unwatch 1', 'Star 0', and 'Fork 0'. Below this is a navigation bar with 'Code', 'Issues 0', 'Pull requests 0', 'Projects 0', 'Wiki', 'Security', 'Insights', and 'Settings'. The main content area is the 'New Issue' form, which includes a 'Title' input field, a 'Write' tab, a 'Preview' tab, and a rich text editor with various formatting options. To the right of the form is a sidebar with settings for 'Assignees' (No one—assign yourself), 'Labels' (None yet), 'Projects' (None yet), and 'Milestone' (No milestone). A red arrow points to the gear icon next to the 'Assignees' setting. Another red arrow points to the 'Submit new issue' button at the bottom right of the form. A text box next to this arrow contains the text 'Clique aqui para criar.'

# Projects

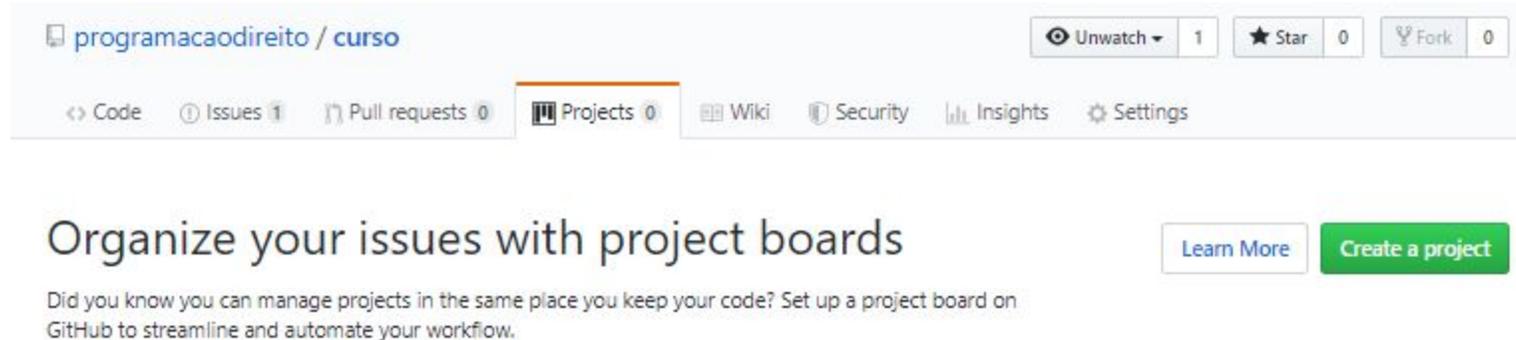
## O que é?

- é uma funcionalidade do Github que permite organizar as issues em Projects Boards.
- Você pode adicionar issues ao quadro e priorizá-los.
- O project board classifica as issues em colunas por status, como "To do", "In Progress" e "Done"

# Projects

## Criando um projeto no Github:

1. Acessar aba "Projects" e clicar em "Create a project":



# Projects

## Criando um projeto no Github:

2. Definir título do projeto e a descrição:

Create a new project

Coordinate, track, and update your work in one place, so projects stay transparent and on schedule.

Project board name

Project board name

Description (optional)

Project template

Save yourself time with a pre-configured project board

Template: None

Create project

Clique aqui para criar.

Clique aqui para escolher template do Project Board

# Projects

## Criando um projeto no Github:

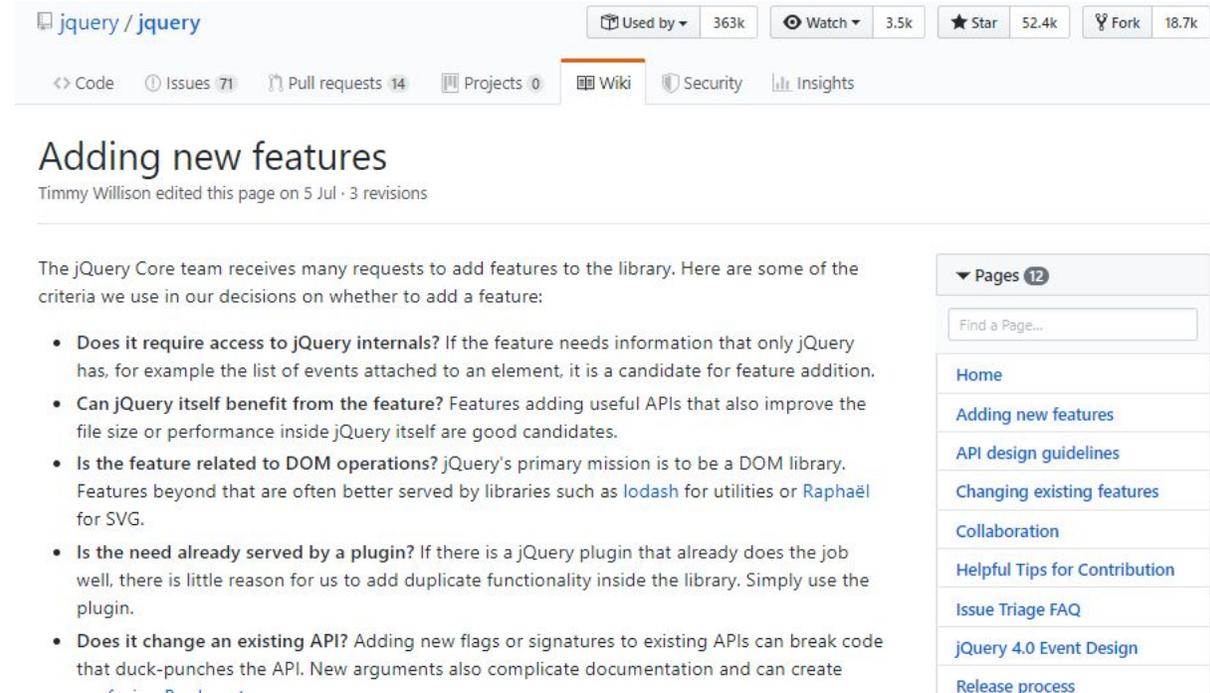
3. Associar issues ao Project Board :
  - a. To do
  - b. In Progress
  - c. Done

A coluna abaixo é contém todas as issues cadastradas (Backlog). para priorizá-los e colocar no sprint, você deve arrastar e soltar na coluna TO DO.



## O que é?

- é a funcionalidade do github, onde hospedamos a documentação do projeto.
- assim outras pessoas podem entender como utilizar e até mesmo contribuir com o seu projeto.
- Markdown



The screenshot shows the GitHub repository page for jQuery, specifically the Wiki section. The page title is "Adding new features" and it was edited by Timmy Willison on July 5th. The page content discusses the criteria for adding new features to the jQuery library. The criteria listed are:

- Does it require access to jQuery internals? If the feature needs information that only jQuery has, for example the list of events attached to an element, it is a candidate for feature addition.
- Can jQuery itself benefit from the feature? Features adding useful APIs that also improve the file size or performance inside jQuery itself are good candidates.
- Is the feature related to DOM operations? jQuery's primary mission is to be a DOM library. Features beyond that are often better served by libraries such as [lodash](#) for utilities or [Raphaël](#) for SVG.
- Is the need already served by a plugin? If there is a jQuery plugin that already does the job well, there is little reason for us to add duplicate functionality inside the library. Simply use the plugin.
- Does it change an existing API? Adding new flags or signatures to existing APIs can break code that duck-punches the API. New arguments also complicate documentation and can create

On the right side of the page, there is a sidebar with a "Pages" section containing 12 items, including "Home", "Adding new features", "API design guidelines", "Changing existing features", "Collaboration", "Helpful Tips for Contribution", "Issue Triage FAQ", "jQuery 4.0 Event Design", and "Release process".

# Fluxo de Trabalho - GIT e Github

## Boas práticas:

- Após criar e clonar o projeto:

1. Assuma uma issue do projeto e altere para “em andamento”;
2. Atualize seu repositório local (git pull);
3. Crie o branch, se necessário, para trabalhar nele;
4. Desenvolva o algoritmo que irá resolver o problema;
5. Adicionar e commitar os arquivos criados;
6. Enviar os arquivos ao repositório remoto (git push);
7. Crie a documentação na wiki.



# Atividades

## Dia 09 – Atividades



## *Breaking the Law*

Copyright 2019 Roberto Novaes ( [www.robertonovaes.com.br](http://www.robertonovaes.com.br) )

Copyright 2019 SiLEX Sistemas ( [www.silexistemas.com.br](http://www.silexistemas.com.br) )

Atribuição-NãoComercial-CompartilhaIgual 4.0 Internacional



Realização:



U F *m* G

Apoio:

**MARCELO CANDIOTTO** **MCSA**  
Sociedade de Advogados



ASSOCIAÇÃO  
BRASILEIRA DE  
LAWTECHS &  
LEGALTECHS