



# OFICINA PRÁTICA CURSO DE GITHUB / GIT

OADS NM1AB-IESB-2024

# Alguns avisos...

- Serão **4h** de Curso
- Teremos uma pausa de **10 minutos**
- Ou cada **1h** ou a cada 2h (vocês irão decidir 😊)
- Em caso de dúvida, fale no momento da dúvida
- Caso se perca, acompanhe a gravação depois 😊
- **Certificado** só para quem permanecer até o final



# QUEM SOU EU?

 Designer / Engenheiro

# Sérgio Silva

 [in/sergiobslva](https://www.linkedin.com/in/sergiobslva)

**para o mundo**

Preto, LGBTQIAPN+, periférico, filho de mãe solteira, mentor de carreiras

**outras informações**

**Graduando** em Análise e Des. de Sistemas



**Engenheiro de Software**  UnB

**6 anos** como UX Designer  SONDA<sup>®</sup>  
make it easy

**16 Anos** Mercado de Tecnologia





# O QUE É

*Github / Git*

# Conceitos

- Github
- Outras plataformas: Github Enterprise (Nuvem Privada)
- Git, SVN
- Outros serviços: Bitbucket (Atlassian)

# PARA QUE SERVE?

*Github / Git*

# Você usa o Git para...

- Armazenar os seus códigos
- Manter projetos de software aberto
- Para códigos dos seus cursos
- Como portfólio



# É PRA QUEM...

*É o mini curso de Git?*

# Recomendo para...

- Desenvolvedores
- Estudantes que mantém códigos para alguma coisa
- Quem tem curiosidade como é a carreira de Dev
- Professores, Designers, Frontends, Backends, Full Stacks
- Curiosos e Profissionais de outras áreas...

# O que você vai aprender aqui hoje?

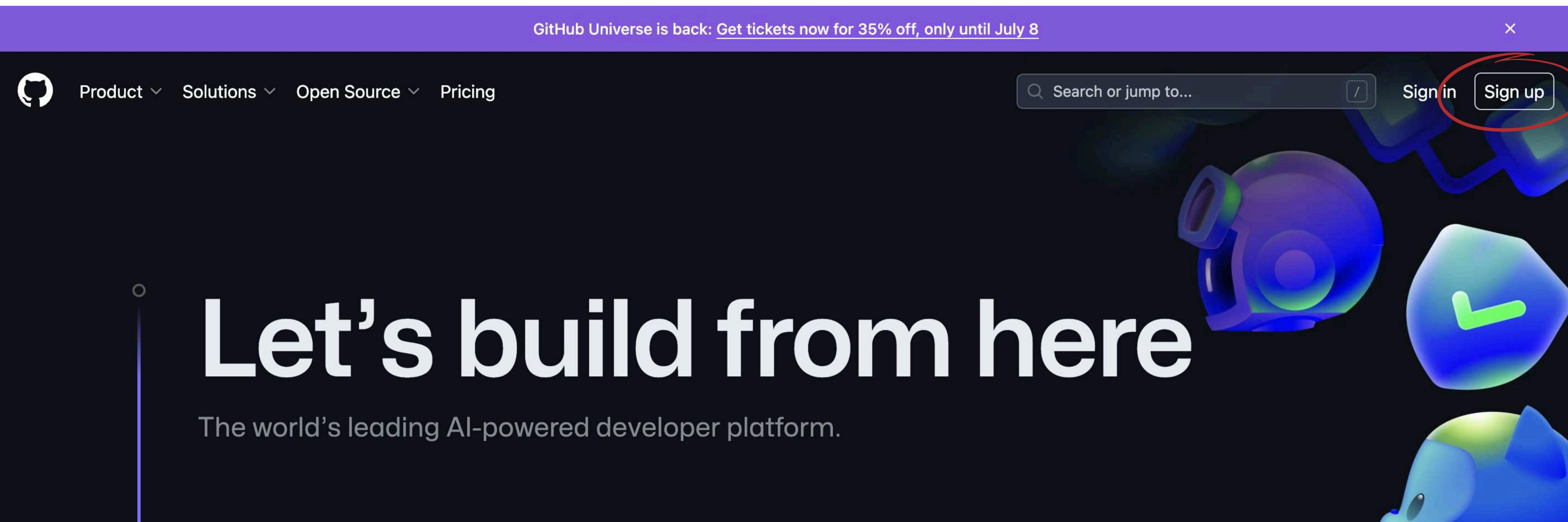
- Criar sua conta no [Github](#)
- Entender o que é Git e o que é Github
- Aprender comandos básicos de Git
- Manter o seu repositório com comandos
- Trabalhar sozinho ou em grupo
- Manter seus trabalhos organizados
- Só aprenderemos o caminho FELIZ!

# VAMOS COMEÇAR

*Criando conta no Github*

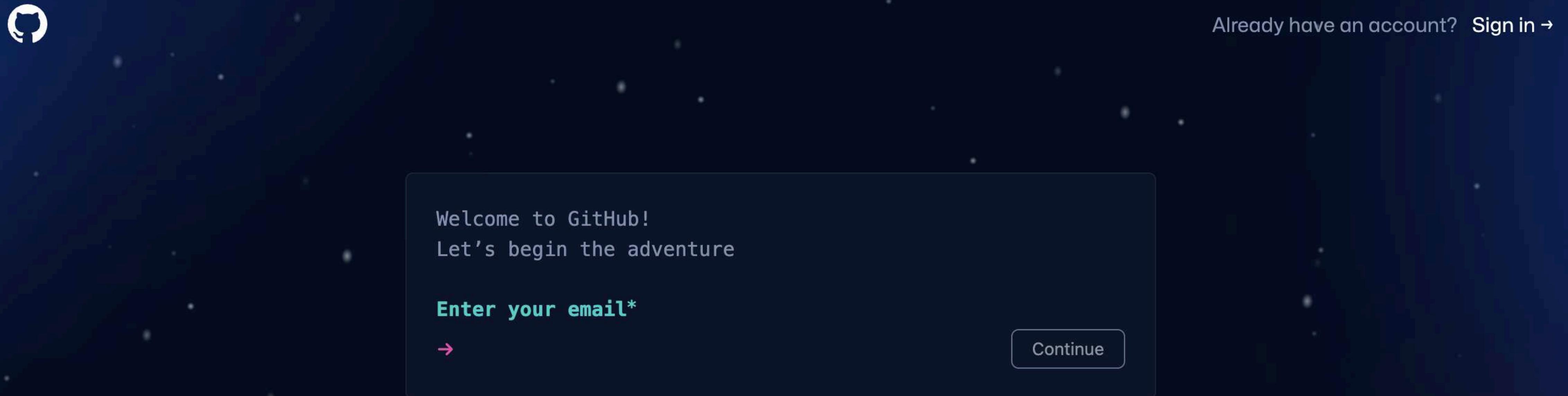
# Criando Conta

- Acesse [github.com](https://github.com) / Clique em **Sign up**



# Criando Conta

- Acesse [github.com](https://github.com)



# INSTALAÇÃO

*Github Terminal*

# Download terminal

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

The screenshot shows the official Git website at <https://git-scm.com/>. The main navigation bar includes links for "About", "Documentation", "Downloads", and "Community". The "Downloads" section is highlighted. It features a large image of a Mac monitor displaying the latest source release version 2.45.0. Below the monitor, there are download links for "macOS", "Windows", and "Linux/Unix". A note states that older releases are available and the Git source repository is on GitHub. The "GUI Clients" section provides information about built-in tools like git-gui and gitk, and links to external clients. The "Logos" section offers various Git logos in PNG and EPS formats.

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

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](#).

## Downloads

macOS Windows Linux/Unix

Latest source Release  
**2.45.0**  
[Release Notes \(2024-04-29\)](#)

Download for Mac

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 →](#)

**Logos**

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

Git via Git

# Passo 1: Instalação

1. Abra o arquivo baixado
2. Instale (Next > Next)
3. Ele habilitará o terminal linux dentro do seu sistema operacional
  - a. IMPORTANTE ESCLARECER (Sobre diferentes S.O.)

# Passo 2: Configurando seu usuário

1. Vá ao seu terminal
2. Digite os comandos:
  - a. git init
  - b. git config --global user.name "**seu nick no github**"
  - c. git config --global user.email **fulanodetal@exemplo.br**



**10 MIN.**

*Yamos Descansar?*

# ORGANIZAÇÃO

*Github / Git*

# Por que você precisa de uma?

- Para separar os seus códigos
- Segmentar os seus projetos



Sérgio Bezerra da Silva

sergiobsilva · he/him

Atualmente UX Research, mas um profissional com competências em UX e UI Design e experiência em criação de produtos digitais.

[Edit profile](#)

6 followers · 4 following

Sonda @ Caixa Econômica

Brasília

sergiobsilva@gmail.com

<https://sergiobsilva.github.io/ux/>

## Achievements



Beta [Send feedback](#)

## Highlights

Developer Program Member

## Organizations



# CRIANDO CHAVE SSH

*Configurando Acesso*

# Por que criar a chave SSH

- Para identificar a seu dispositivo
- Para permitir que seus arquivos tramitem de forma segura
- Sem precisar ficar autenticando toda hora
- Para garantir que a sua assinatura estará em  
seus arquivos

# Passo 1: Adicionando a chave no Github

1. Vamos gerar a chave

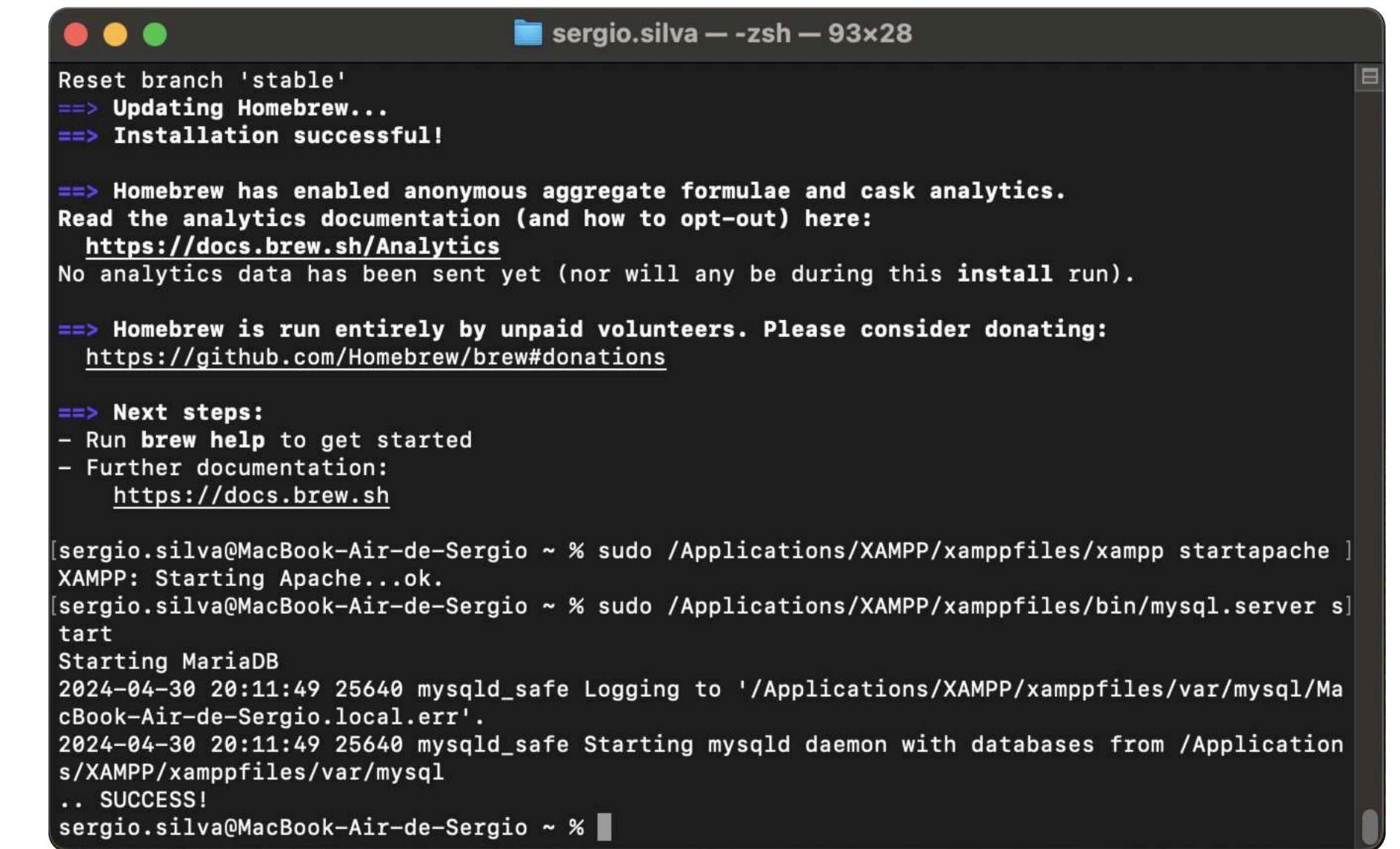
a. Vá até o seu terminal

b. Digite o comando:

i. **ssh-keygen**

2. Agora, vamos navegar até a

pasta para pegarmos essa chave  
e colocarmos no Github!



```
Reset branch 'stable'
==> Updating Homebrew...
==> Installation successful!

==> Homebrew has enabled anonymous aggregate formulae and cask analytics.
Read the analytics documentation (and how to opt-out) here:
https://docs.brew.sh/Analytics
No analytics data has been sent yet (nor will any be during this install run).

==> Homebrew is run entirely by unpaid volunteers. Please consider donating:
https://github.com/Homebrew/brew#donations

==> Next steps:
- Run brew help to get started
- Further documentation:
https://docs.brew.sh

[sergio.silva@MacBook-Air-de-Sergio ~ % sudo /Applications/XAMPP/xamppfiles/xampp startapache ]
XAMPP: Starting Apache...ok.
[sergio.silva@MacBook-Air-de-Sergio ~ % sudo /Applications/XAMPP/xamppfiles/bin/mysql.server s]
tart
Starting MariaDB
2024-04-30 20:11:49 25640 mysqld_safe Logging to '/Applications/XAMPP/xamppfiles/var/mysql/Ma
cBook-Air-de-Sergio.local.err'.
2024-04-30 20:11:49 25640 mysqld_safe Starting mysqld daemon with databases from /Application
s/XAMPP/xamppfiles/var/mysql
.. SUCCESS!
sergio.silva@MacBook-Air-de-Sergio ~ % ]
```

# Passo 2: Adicionando a chave no Github

1. Vamos gerar a chave
  - a. No seu repositório logado, vá em **Settings**
  - b. Procure do lado esquerdo da tela **SSH and GPG Keys**
  - c. Clique no botão **New SSH Key**
  - d. Adicione os dados da chave que você criou anteriormente

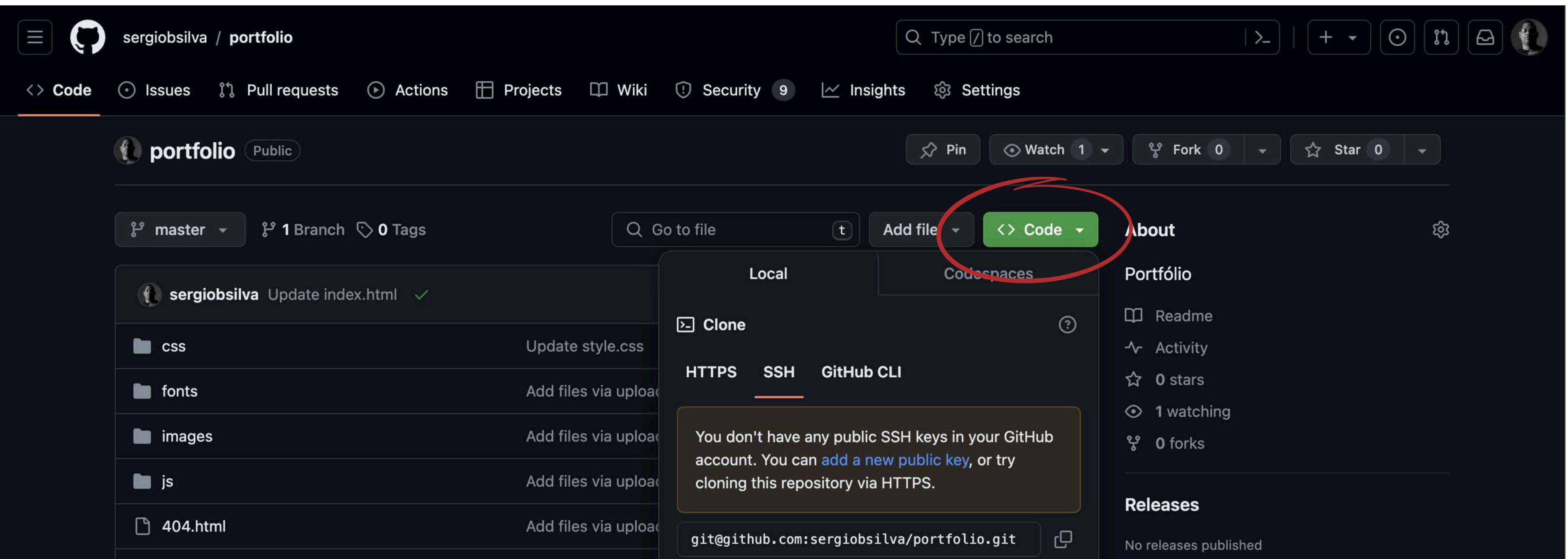


# CLONANDO

*Repositório no Github*

# Efetuando cópia do repositório

1. Escolha o repositório que você deseja copiar o código
2. Vá até o botão verde e clique para ver as opções de cópia



# Por que criar a chave SSH

- Para identificar a seu dispositivo
- Para permitir que seus arquivos tramitem de forma segura
- Sem precisar ficar autenticando toda hora
- Para garantir que a sua assinatura estará em  
seus arquivos

# CICLO DE VIDA

*De um projeto dentro do git*

# Passo 1: Criando e Clonando

- Vamos primeiro criar um repositório no git
- Agora vamos copiar a URL via SSH (como já aprendemos)
- Clone o repositório com **git clone** [url copiada do ssh]
- Vamos modificar algo em nosso projeto

# Passo 2: Fluxo de Upstream

1. Vamos ver o que mudou com **git status**
2. Vamos adicionar o que foi modificado com **git add . -A**
3. Agora vamos adicionar os arquivos na área remota  
com o comando **git commit -m "mensagem do commit"**
4. Estamos no final do fluxo, é hora de mandar o que está  
em nossa máquina, para o repositório REMOTO com o  
comando **git push**

# EXTRA

1. Vamos agora alterar algo no repositório remoto  
e depois solicitar estas modificações com **git pull**

# MULTIVERSO GIT

*Vamos criar uma nova linha do tempo*



# Criando outro nó...

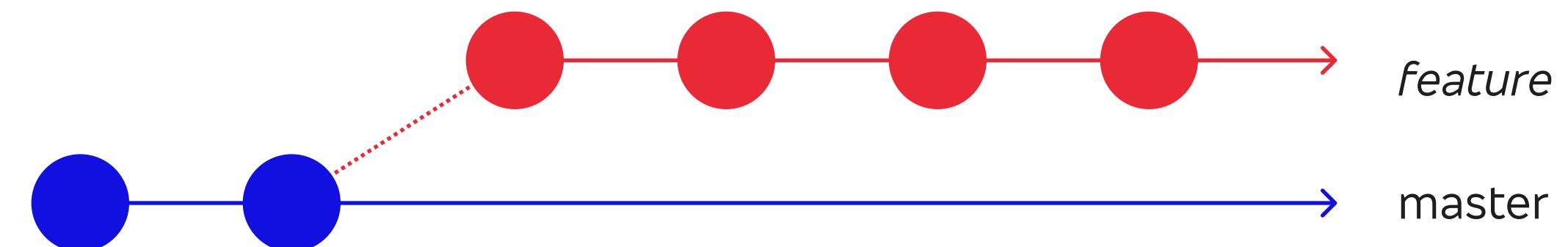
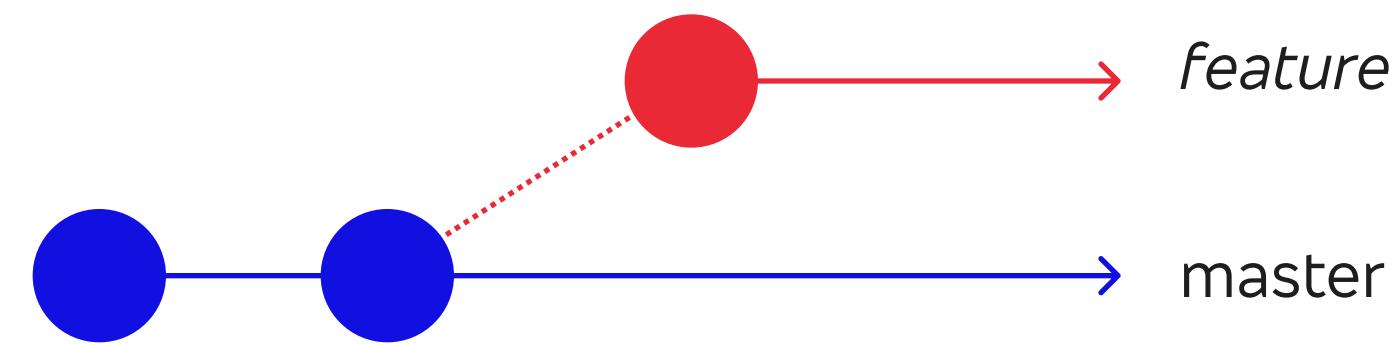
1. Utilizamos esta opção quando precisamos criar uma outra versão do meu projeto, mas com características do projeto original.
2. Também é utilizado para manter outras *features* do produto, mas que não afetem a versão original do projeto.

# Criando outro nó...

1. Para criar uma nova branch (como é chamada uma outra bifurcação do projeto) basta usar o comando  
**git checkout -b nome-da-nova-branch**
2. Imediatamente a nova branch será criada e o git irá mudar para a nova branch!
3. Vamos agora saber como esta branch nova fica no git?
4. [REPITA TODO O CICLO DE VIDA QUE JÁ FALAMOS ANTES]

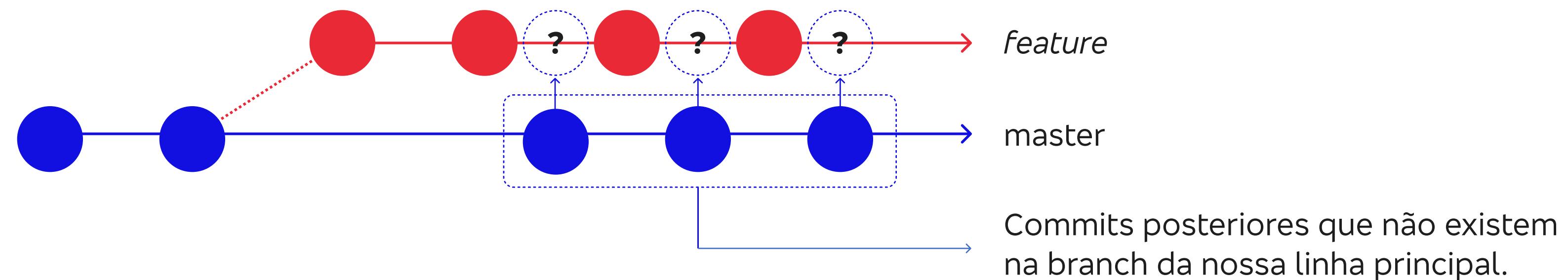
# Gráfico de funcionamento do git

1. Gráfico de quando uma nova branch é criada:



# Gráfico de funcionamento do git

1. Gráfico de commits na timeline de um projeto:



# MERGE E CONFLITOS

*Github / Git*

# Quando precisamos fazer MERGE

- Quando precisamos levar uma modificação de uma outra **branch** para unir estas modificações a **master**

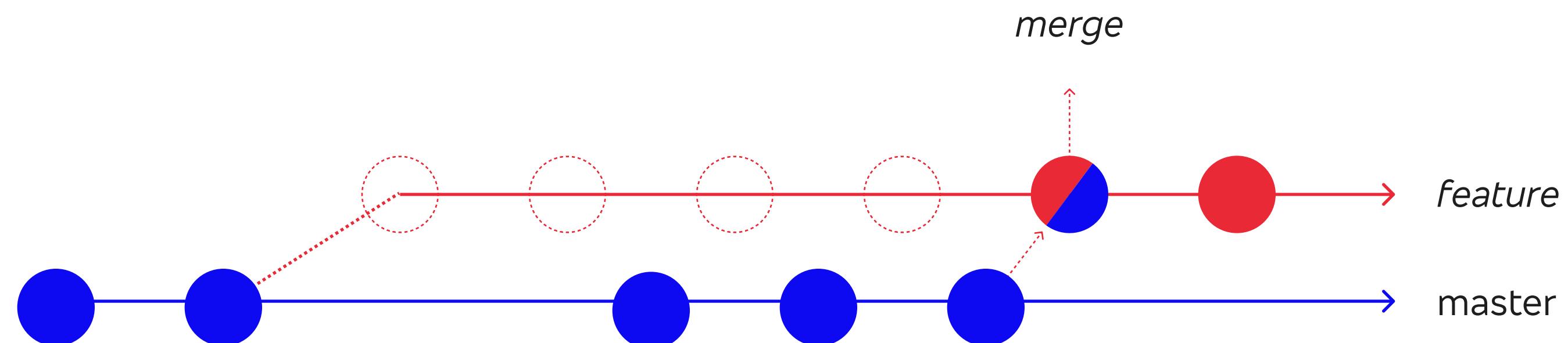
# Antes, CUIDADO!

- Primeiro, observe qual a **branch** que você está
- **ATENÇÃO:** O **merge** é uma operação séria, e a **junção** das informações de uma **branch** pode quebrar a sua aplicação, **então**, use com cuidado!

# Como fazer o Merge

- Confirmando a branch que você está no momento,
- Execute o comando:

**git merge *branch-ira-se-unir-a-master***



# Novidades

- Confirmando a branch que você está no momento,
- Execute o comando:

**git restore** restaure um arquivo ou projeto

**git switch** mesma função do **git checkout -b** (para trocar de branch)

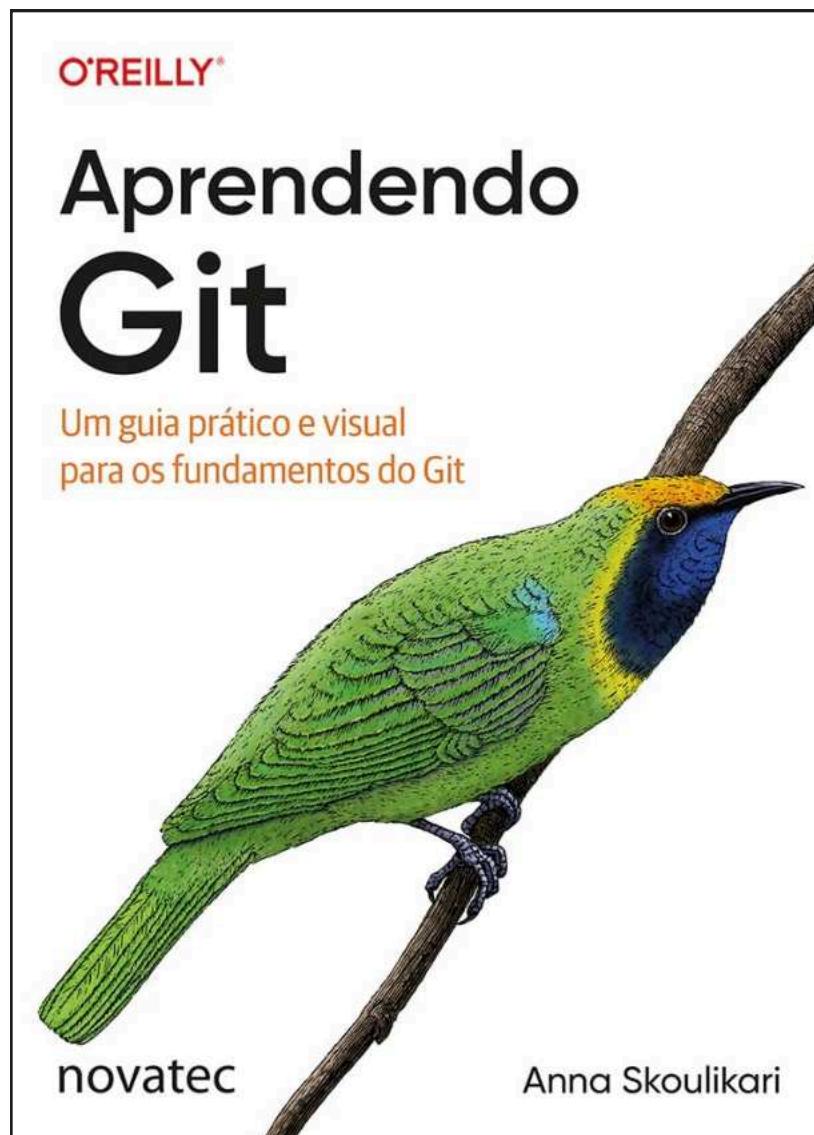
# Resolvendo um conflito

- Após fazer **merge**
- Podem surgir conflitos com a branch **master**
- Vamos então resolver os conflitos :)

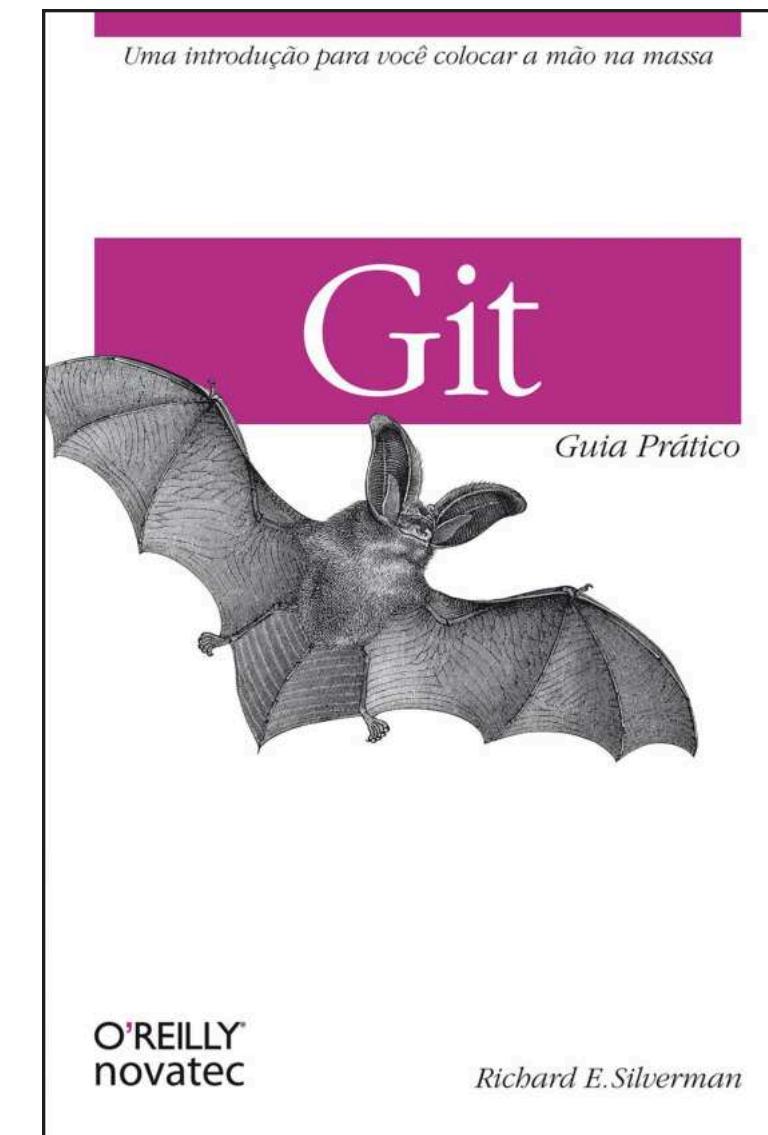
# PARA SABER MAIS

*Sobre Git e Github*

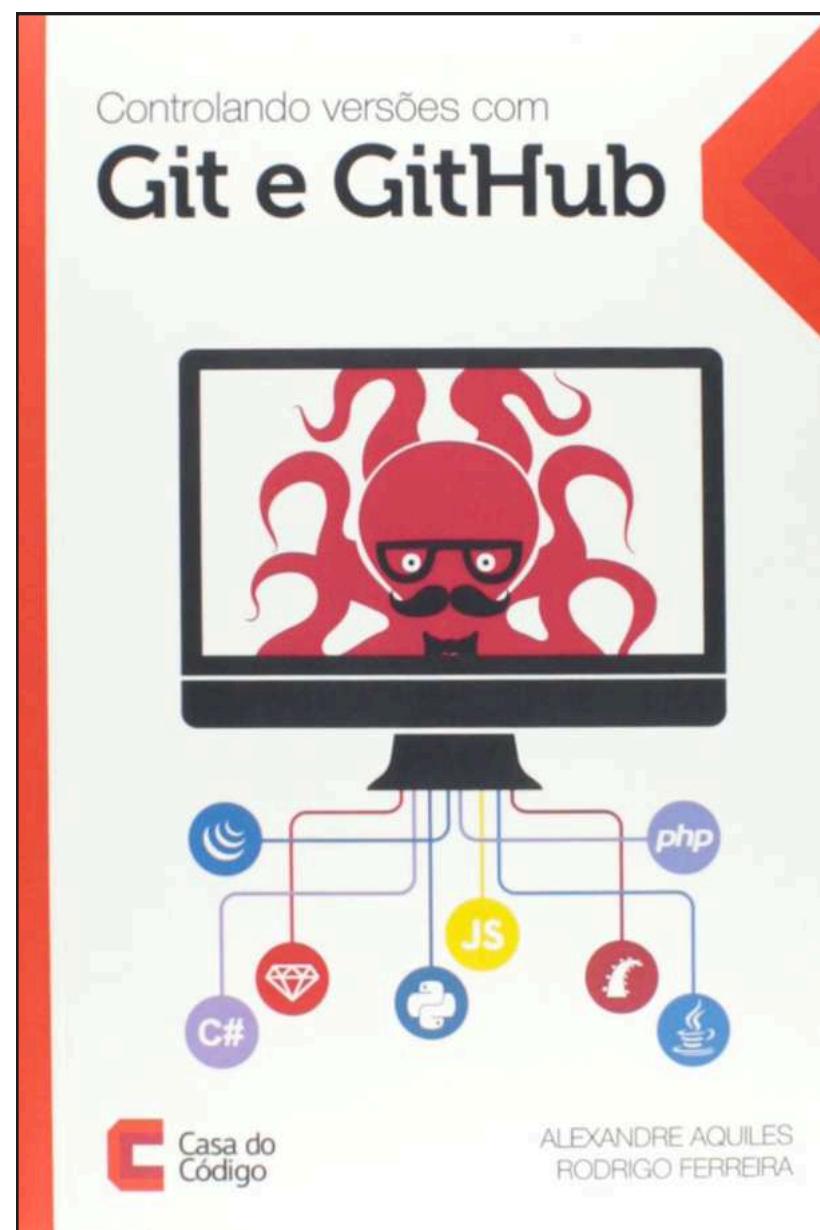
# Livros



**Aprendendo Git: Um  
guia prático e visual  
para  
os fundamentos do Git**  
por Anna  
Skoulikari (Autor)

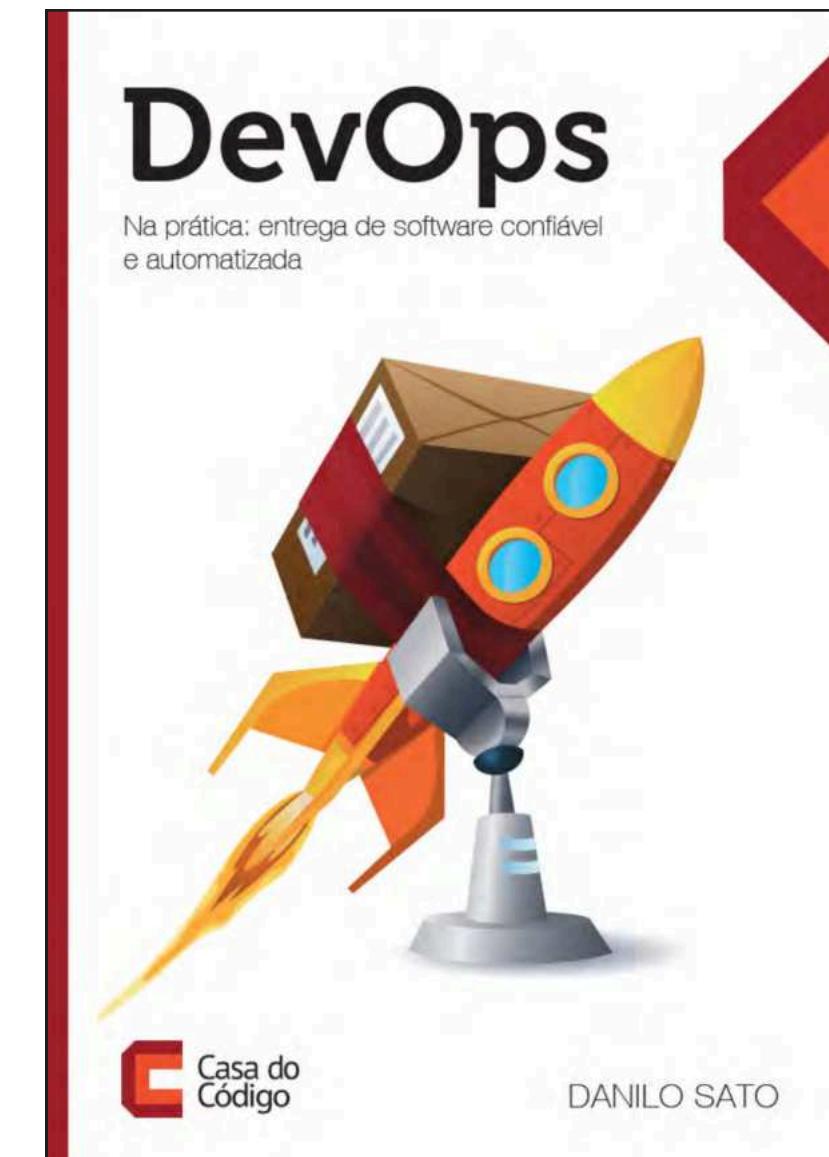


**Git. Guia Prático**  
por Richard E. Silverman



## Controlando Versões com Git e GitHub

por Alexandre Aquiles



## Controlando Versões com Git e GitHub

por Alexandre Aquiles

# Links importantes

1. Site **git - guia prático** ([https://rogerdudler.github.io/git-guide/index.pt\\_BR.html](https://rogerdudler.github.io/git-guide/index.pt_BR.html))
2. Treine comandos git  
[https://learngitbranching.js.org/?locale=pt\\_BR](https://learngitbranching.js.org/?locale=pt_BR)

# Sobre conflitos...



Curso gratuito Git e Github  
#11 - Resolvendo conflitos

# OBRIGADO!

*Professor Marcelo*







# Alguns avisos...

- Nosso curso é **PRÁTICO**, então peço a participação de vocês de forma **sonora** com os microfones quando estivermos construindo os nossos **exercícios!**