Git e GitHub

Felipe P. Coutinho

Roteiro

1. Motivação
2. O que é versionamento?
3. Git e GitHub
4. Mini-guia: GitHub de forma fácil
5. Usando Git e GitHub em conjunto
6. Referências interessantes

Motivação

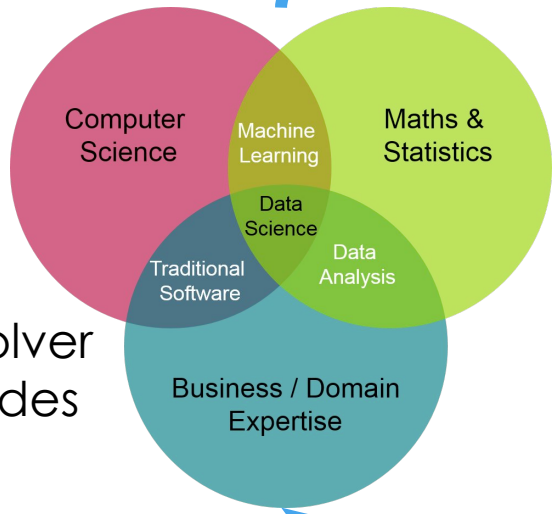
Desenvolver projetos

kaggle

Practicum

SIGMAGEEK

Disponibilizar projetos

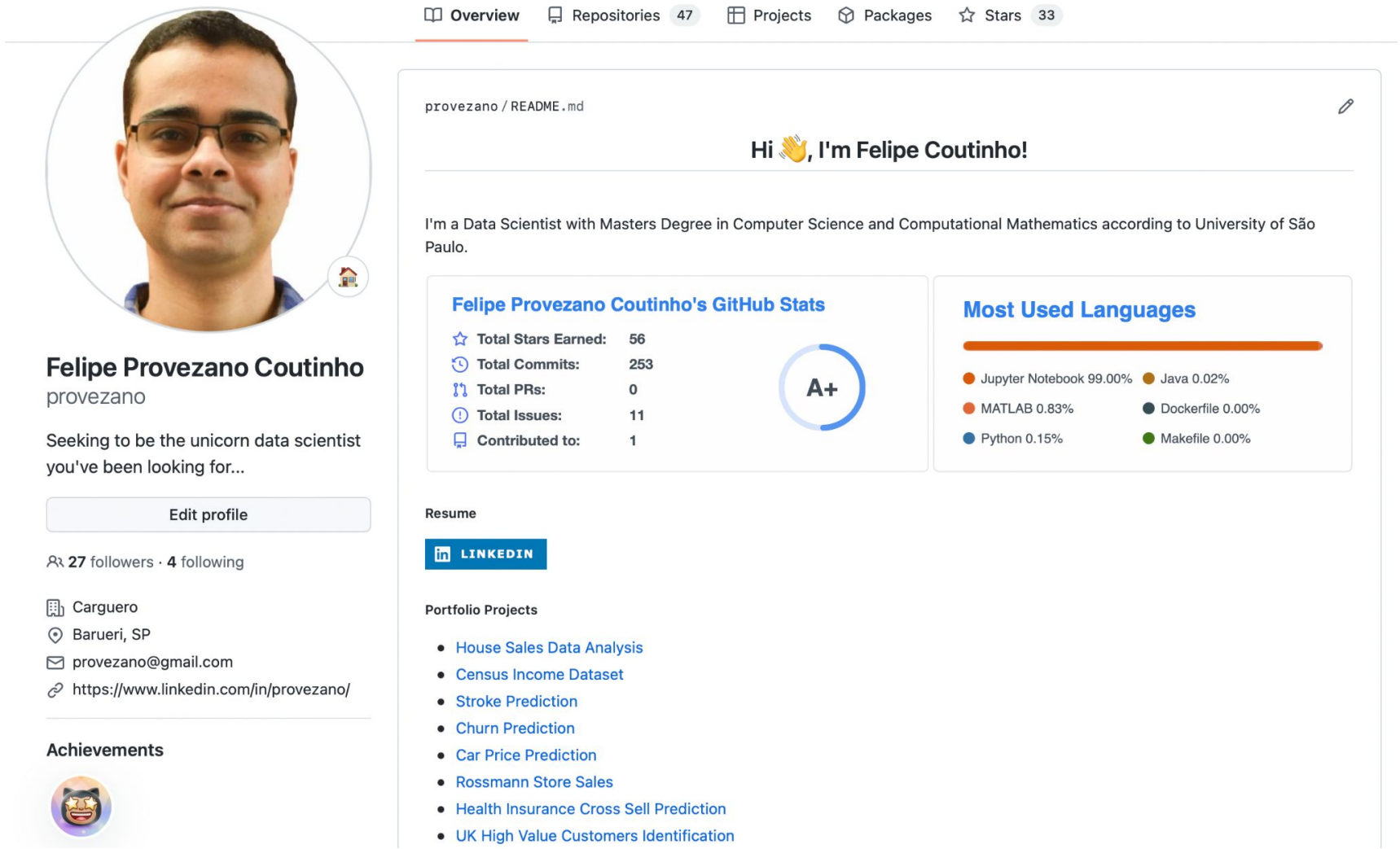


Desenvolver habilidades

Aprendizado cíclico

Divulgar seu trabalho e aplicar para vagas

Onde hospedar meus projetos?



Overview | Repositories 47 | Projects | Packages | Stars 33

Felipe Provezano Coutinho
provezano

Seeking to be the unicorn data scientist you've been looking for...

Edit profile

27 followers · 4 following

Carguero
Barueri, SP
provezano@gmail.com
https://www.linkedin.com/in/provezano/

Achievements

provezano / README.md

Hi 👋, I'm Felipe Coutinho!

I'm a Data Scientist with Masters Degree in Computer Science and Computational Mathematics according to University of São Paulo.

Felipe Provezano Coutinho's GitHub Stats

Total Stars Earned:	56
Total Commits:	253
Total PRs:	0
Total Issues:	11
Contributed to:	1

A+

Most Used Languages

Jupyter Notebook 99.00% · Java 0.02% · MATLAB 0.83% · Dockerfile 0.00% · Python 0.15% · Makefile 0.00%

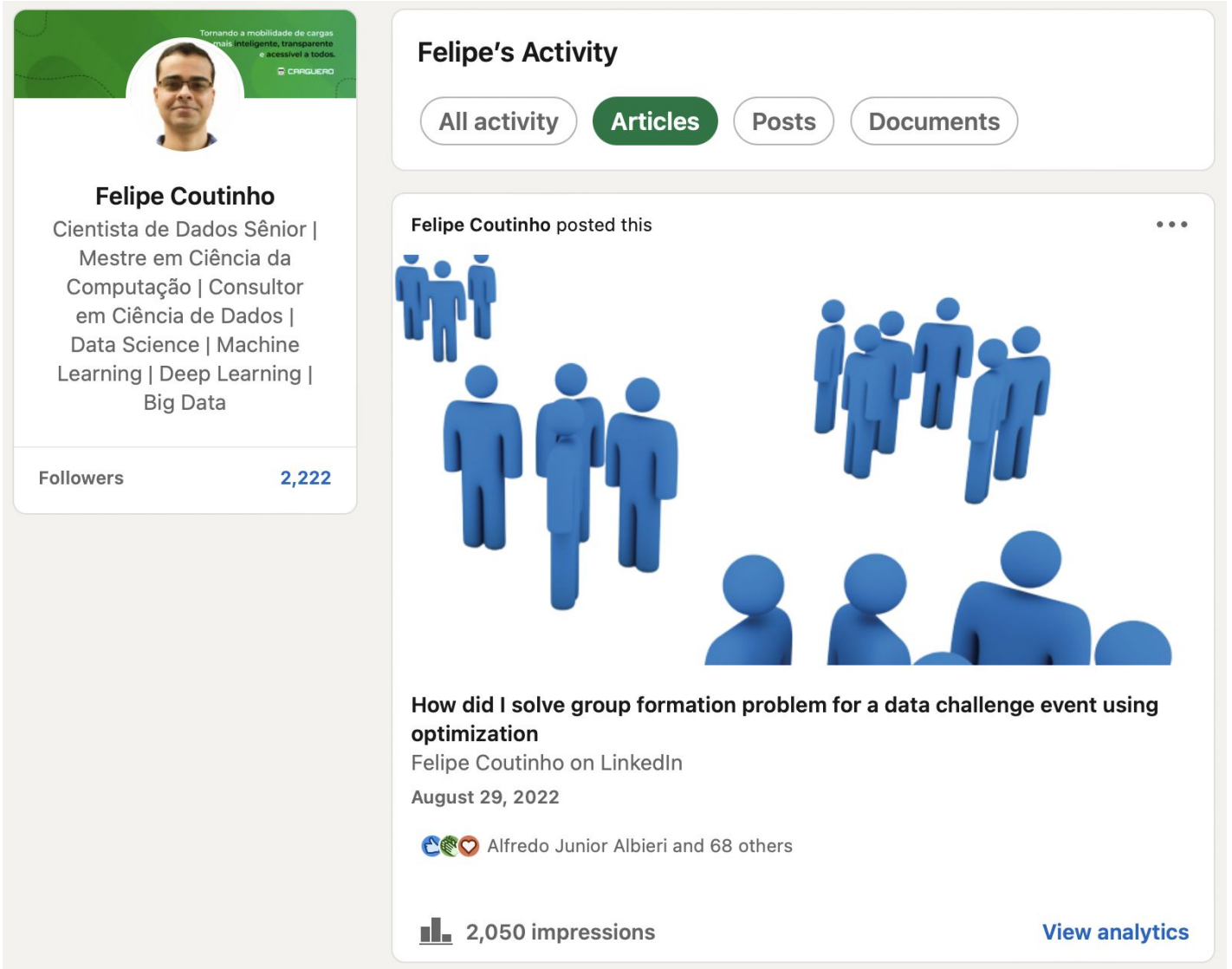
Resume

LINKEDIN

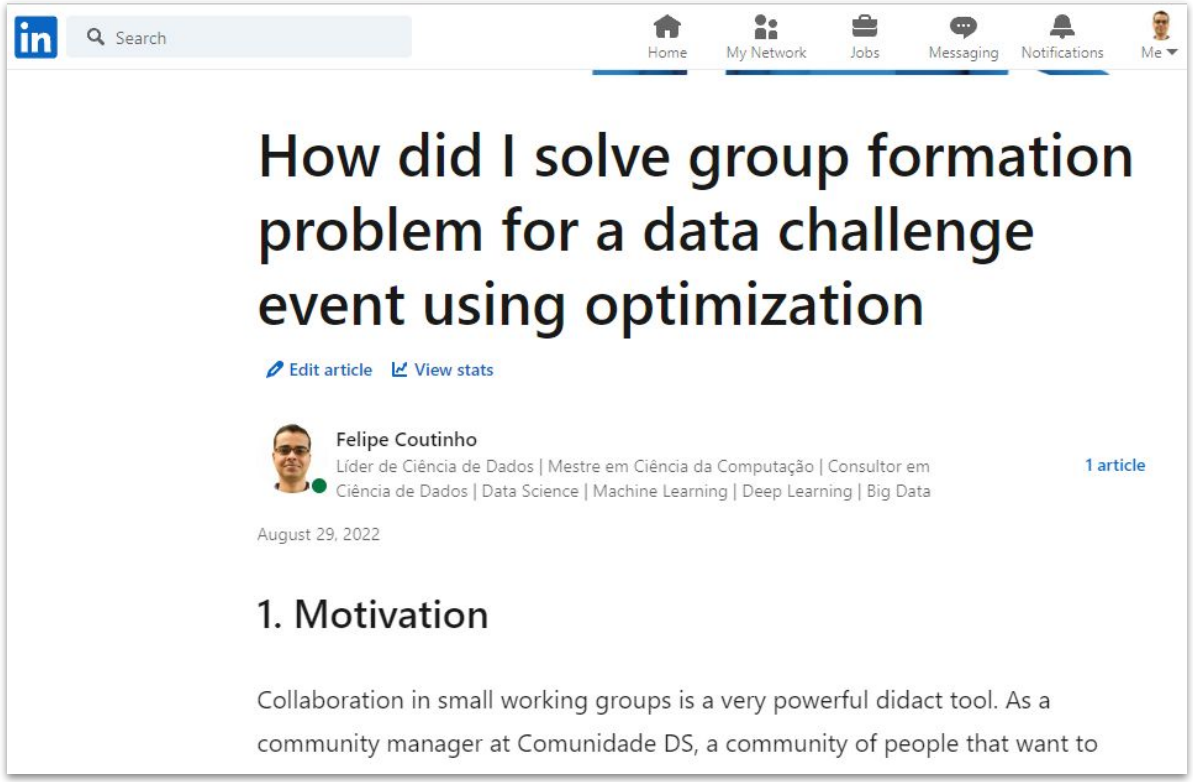
Portfolio Projects

- House Sales Data Analysis
- Census Income Dataset
- Stroke Prediction
- Churn Prediction
- Car Price Prediction
- Rossmann Store Sales
- Health Insurance Cross Sell Prediction
- UK High Value Customers Identification

Onde divulgar meus projetos?

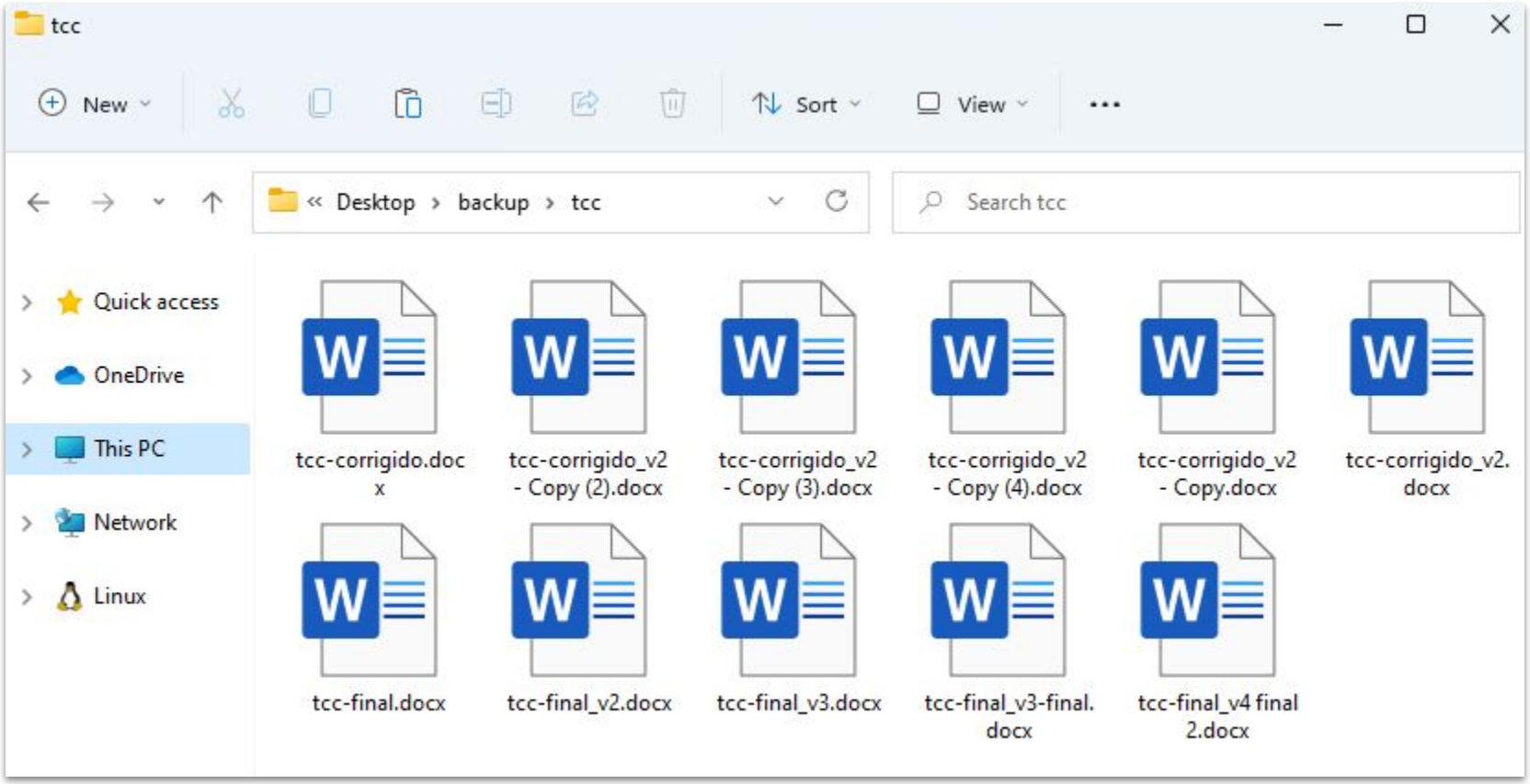


Exemplo prático



[How did I solve group formation problem for a data challenge event using optimization | LinkedIn]

Você já tentou fazer controle de versão de arquivos

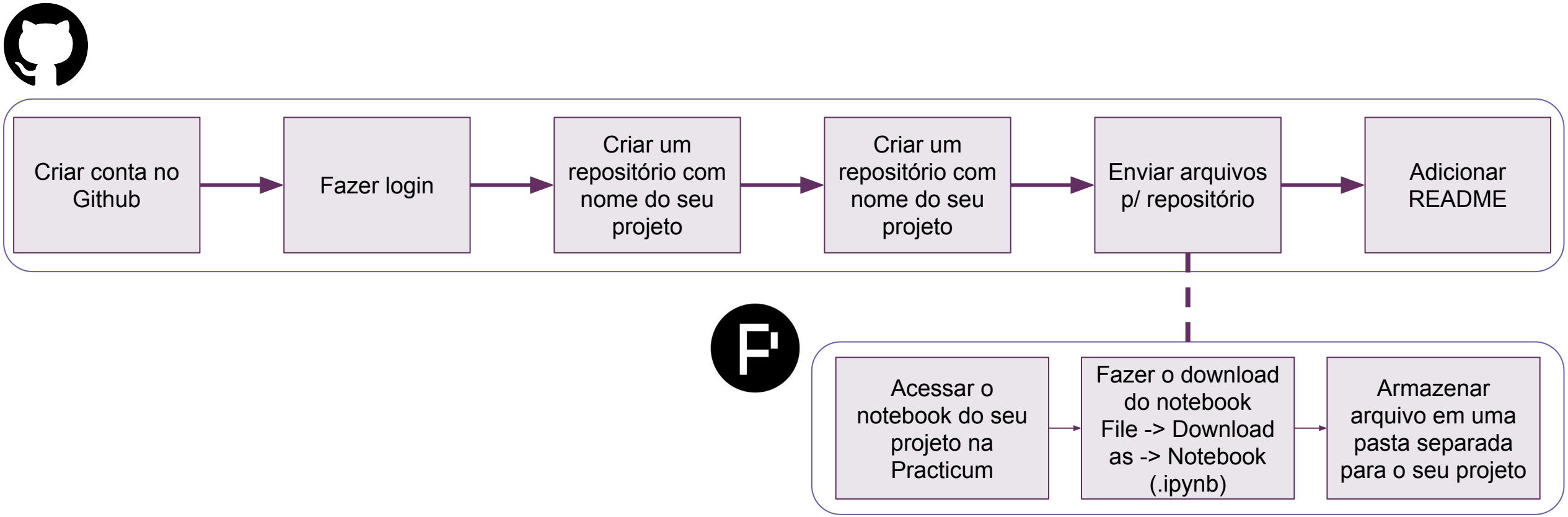


Git e GitHub

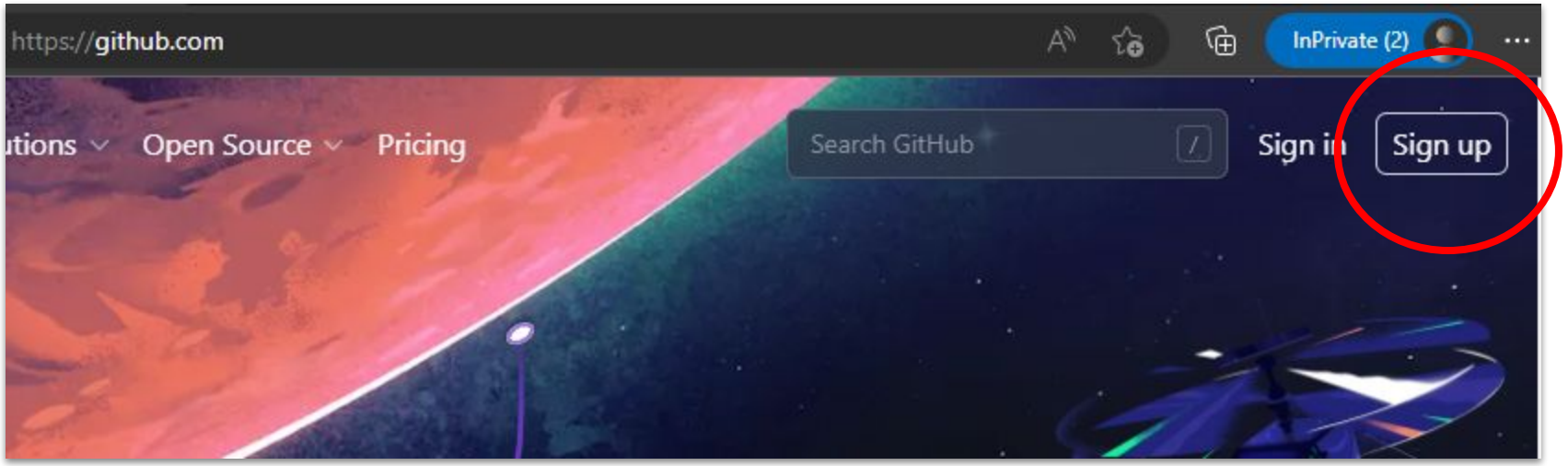
Ferramenta para **controle de versão** de arquivos em **repositórios**.

Plataforma de **hospedagem** de arquivos com *controle de versão* **usando o Git**.

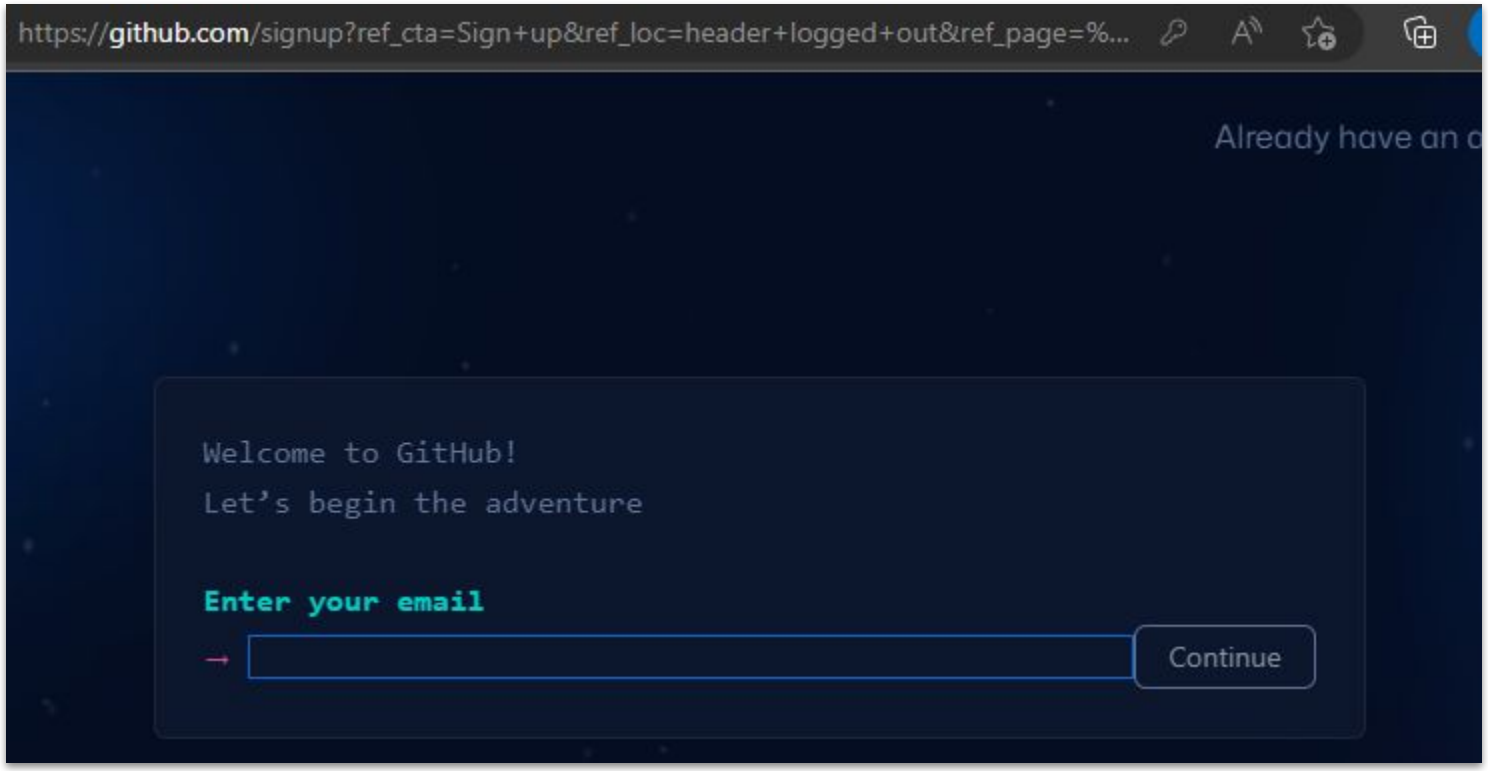
A forma mais fácil



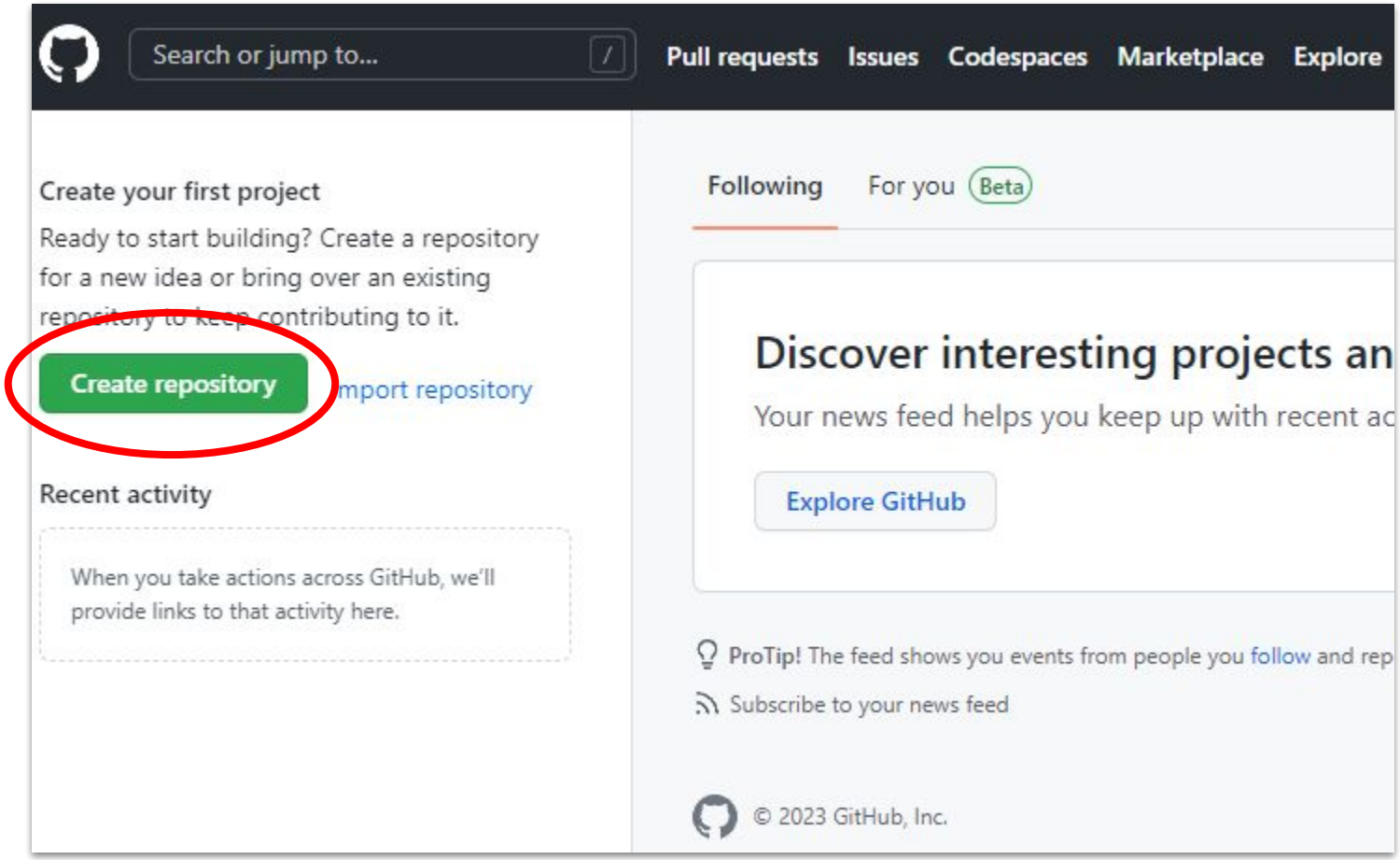
1. Criar conta no GitHub (1/2)



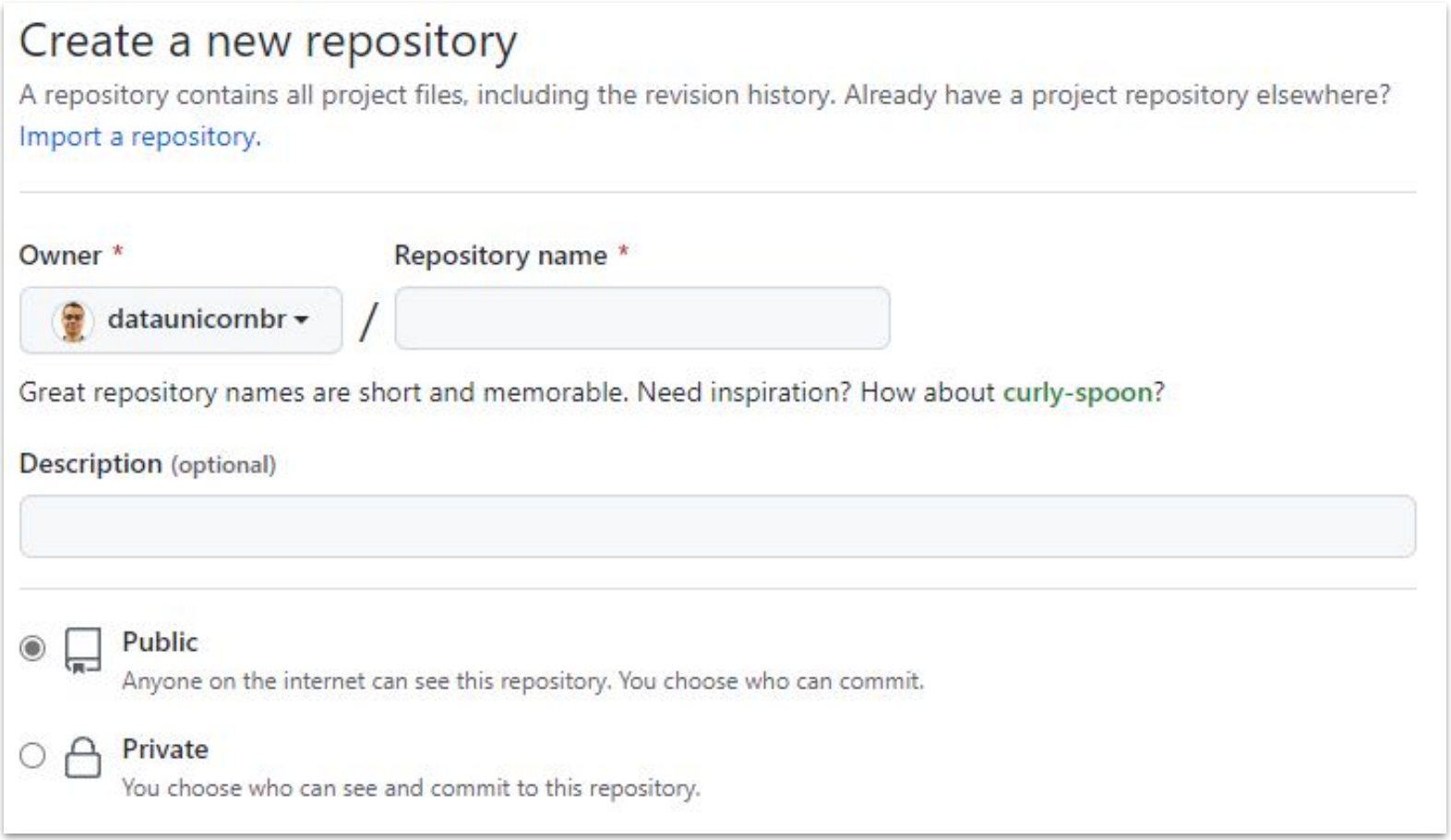
1. Criar conta no GitHub (2/2)



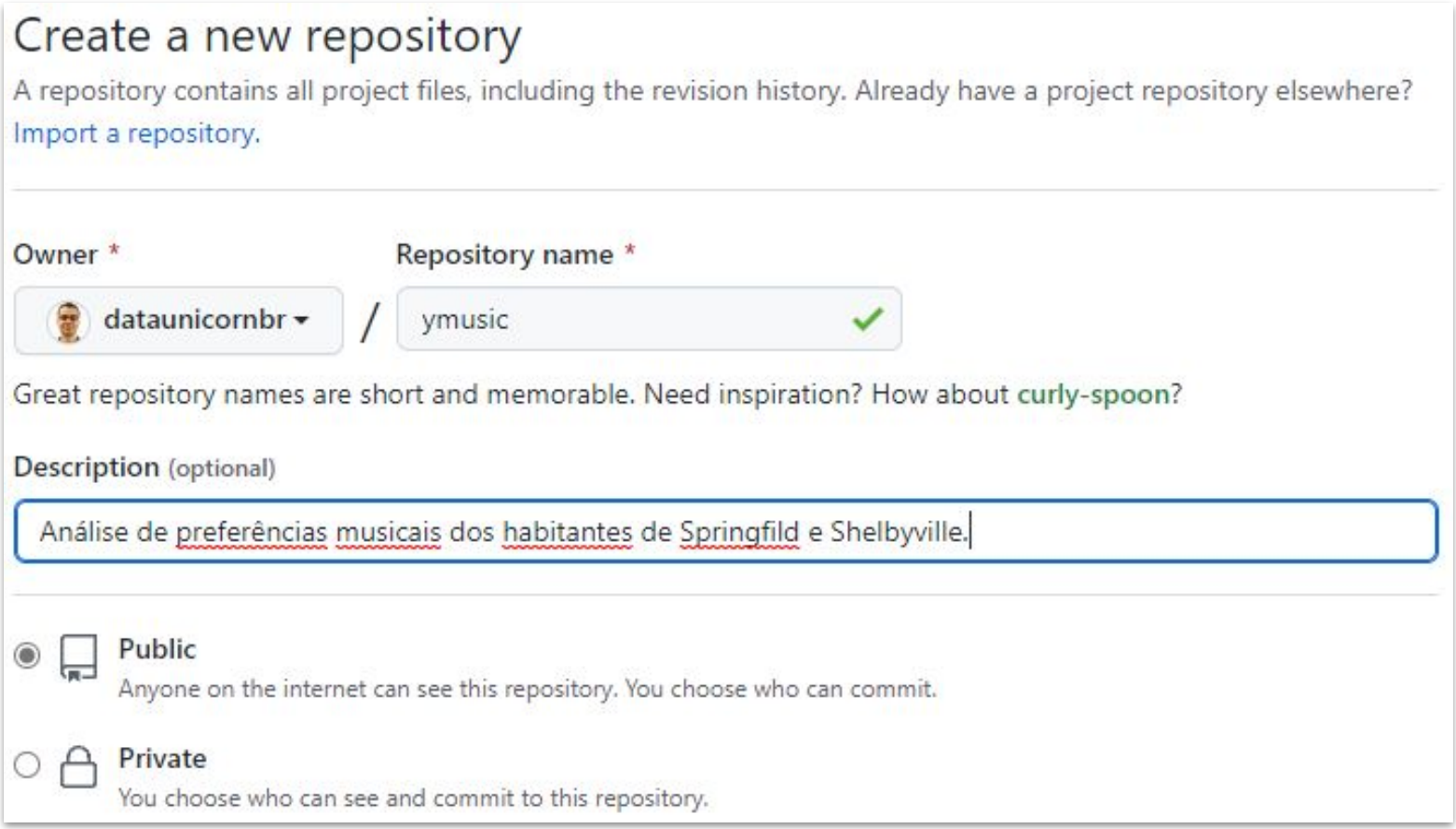
2. Criar repositório no GitHub (1/4)



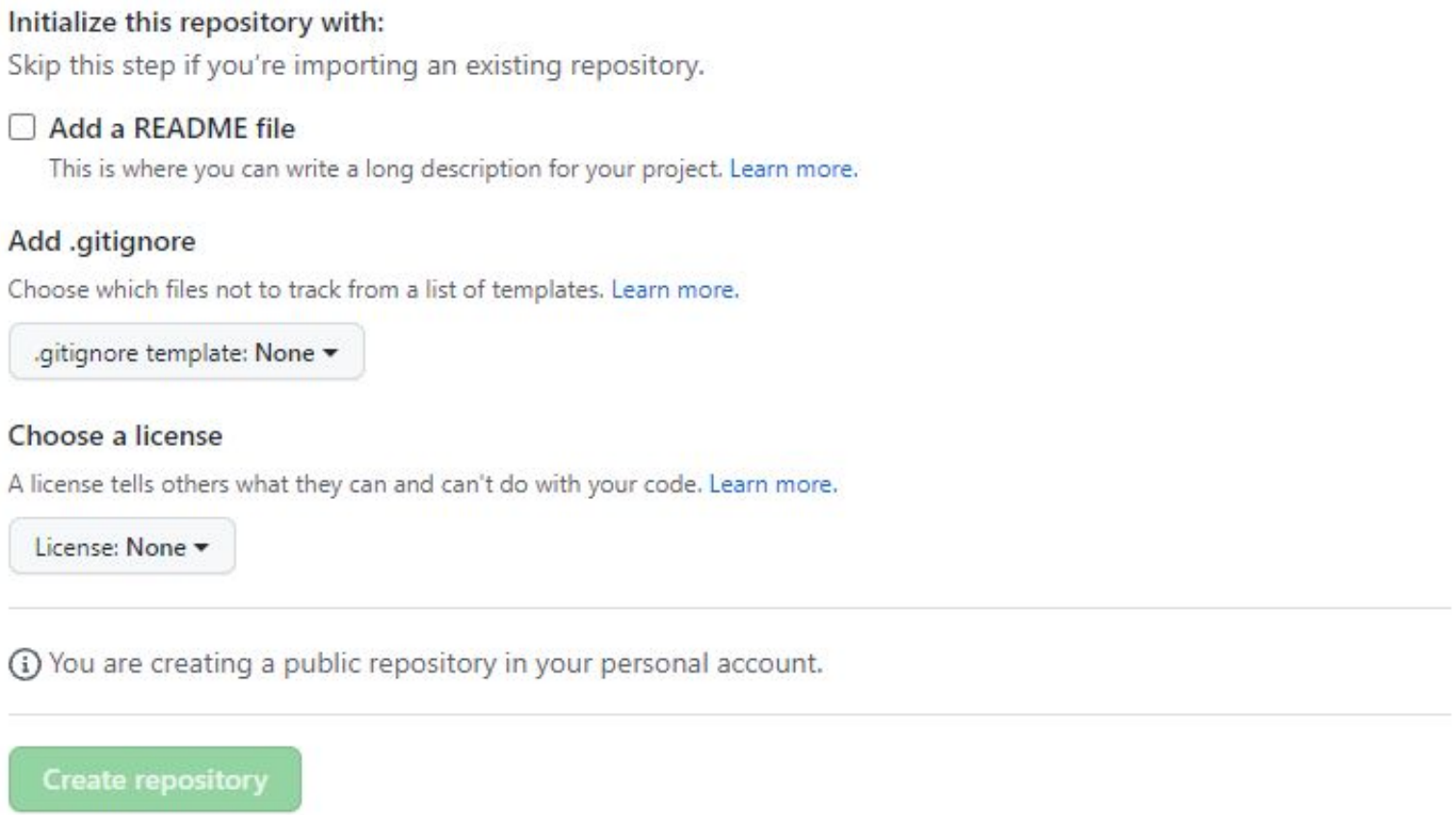
2. Criar repositório no GitHub (2/4)



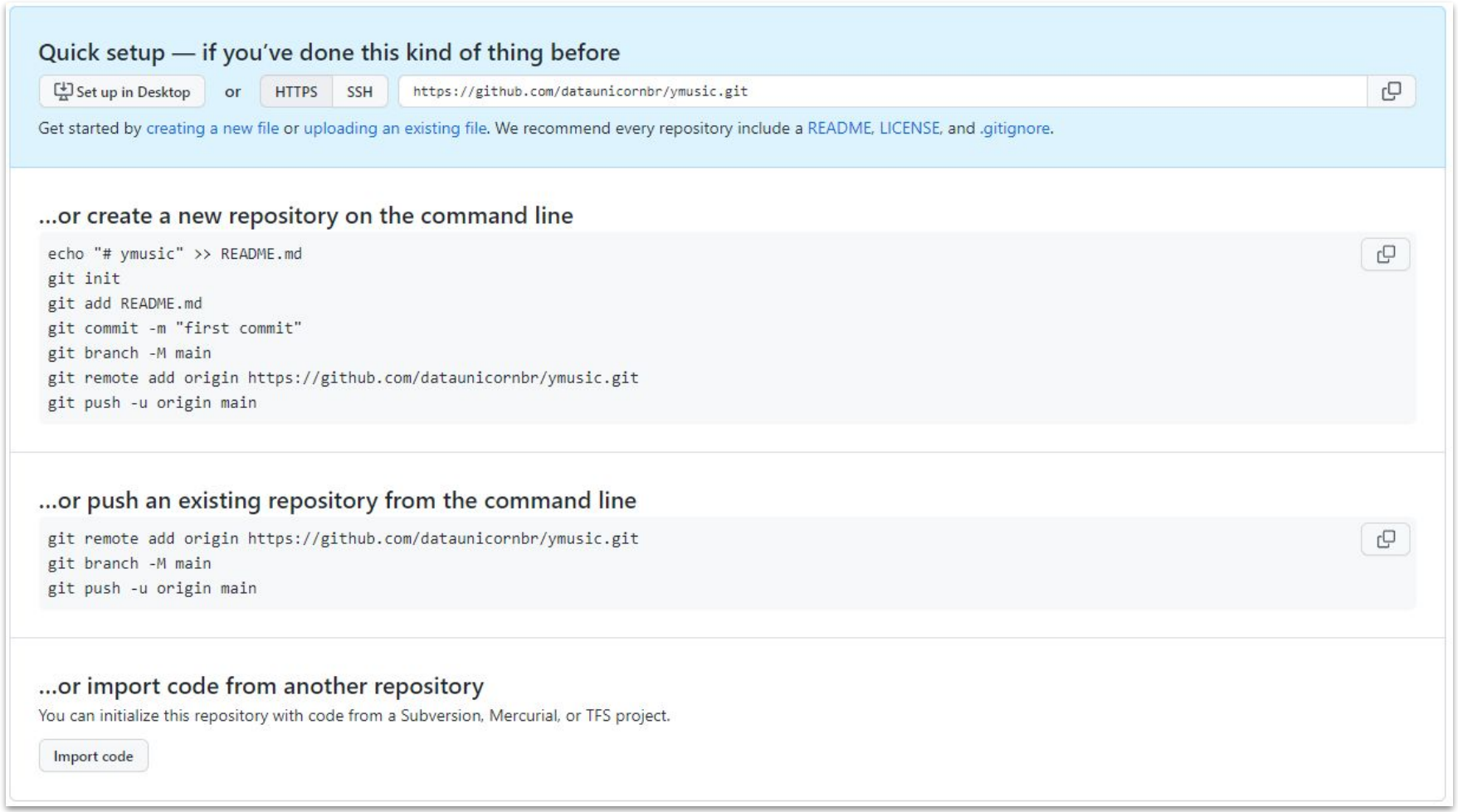
2. Criar repositório no GitHub (3/4)



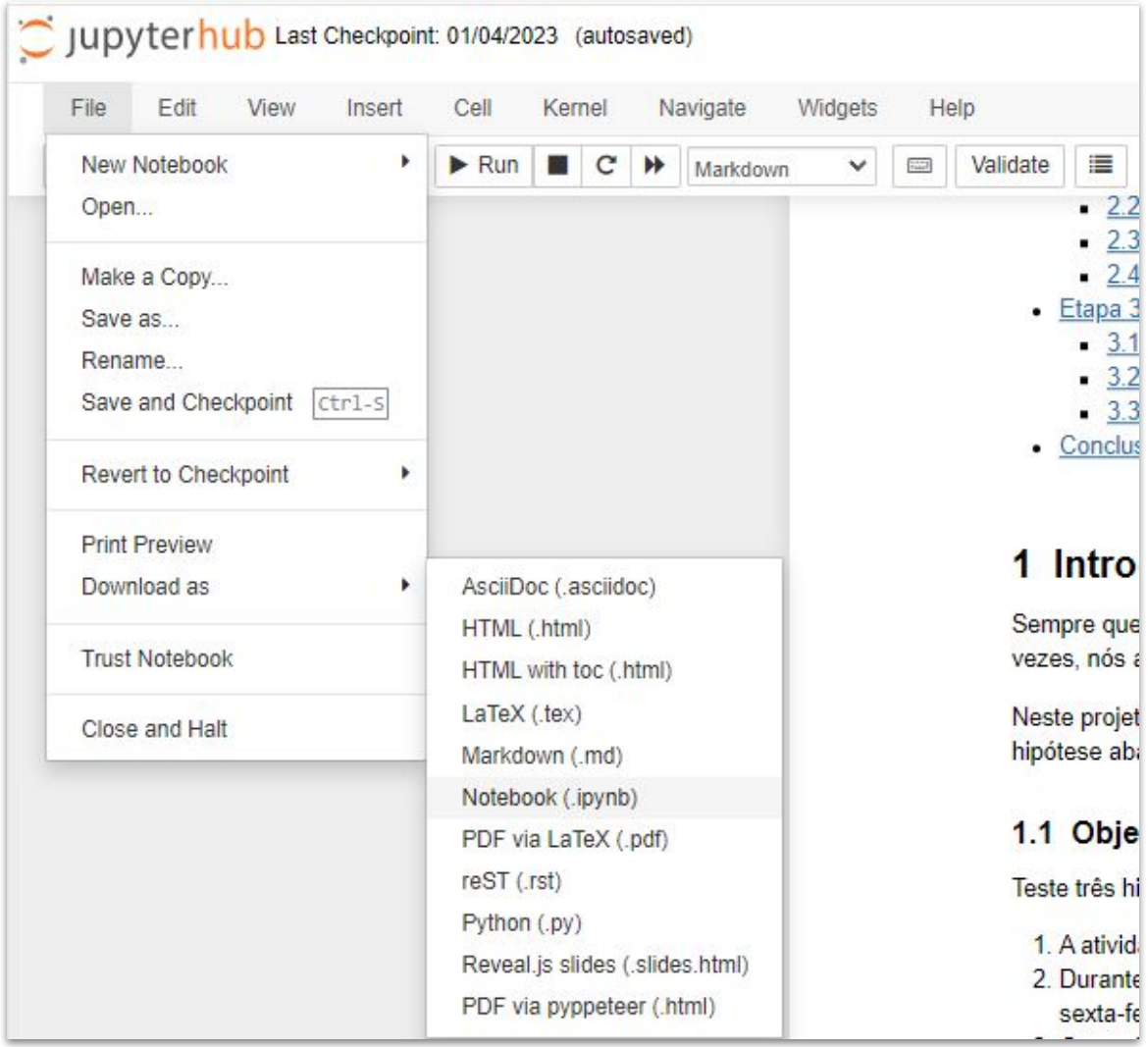
2. Criar repositório no GitHub (4/4)



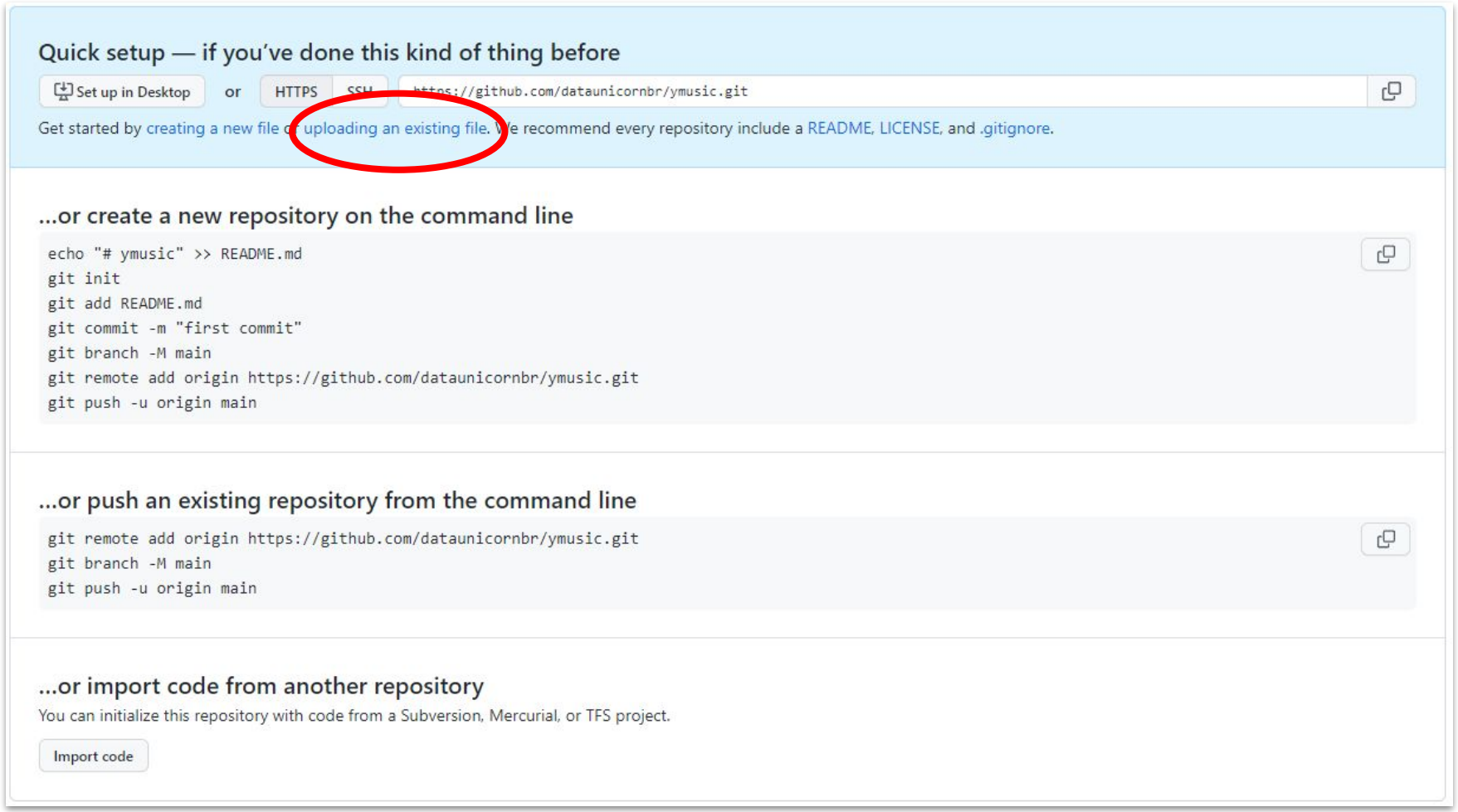
3. Repositório criado



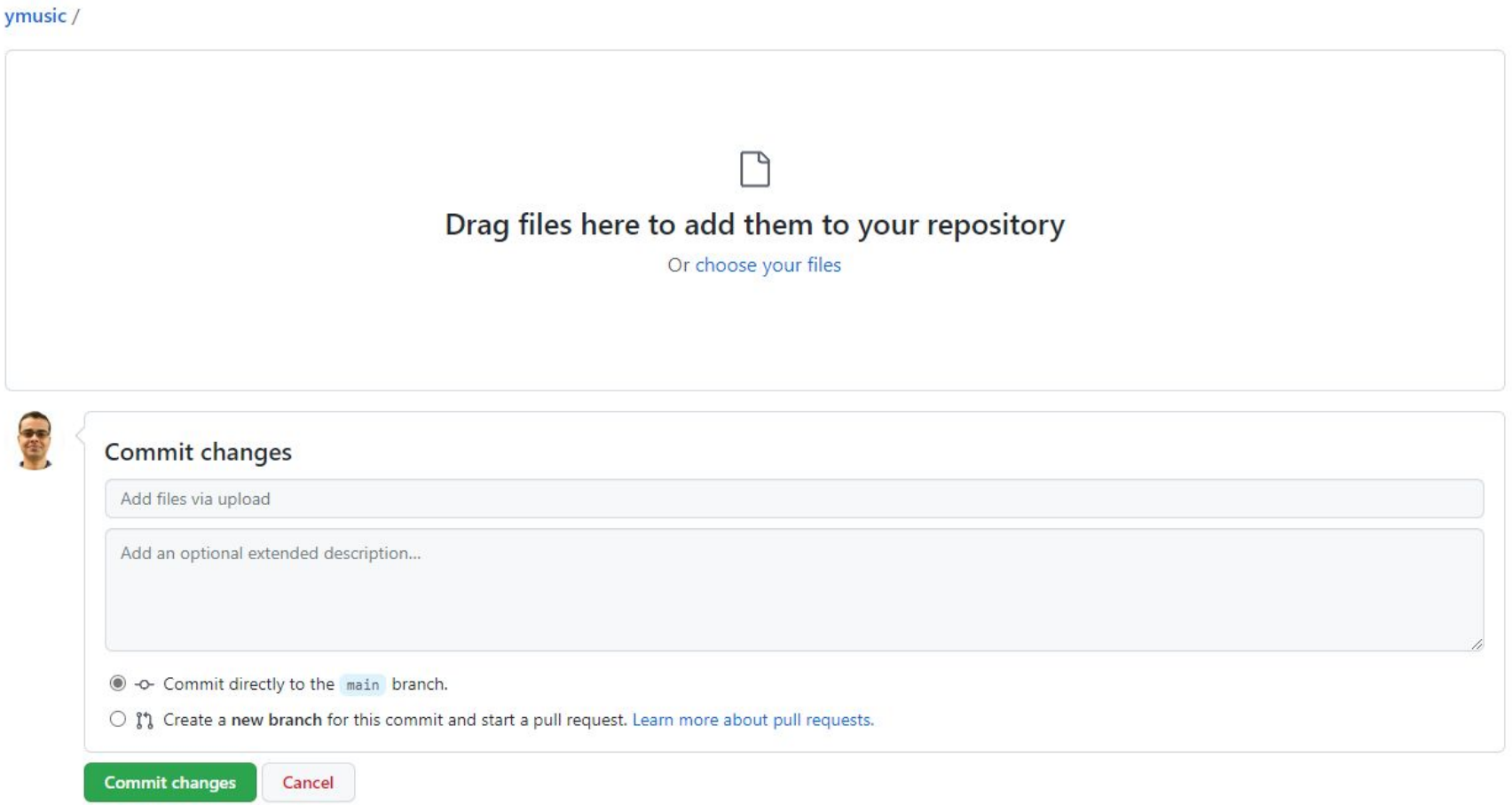
4. Baixar arquivos do projeto



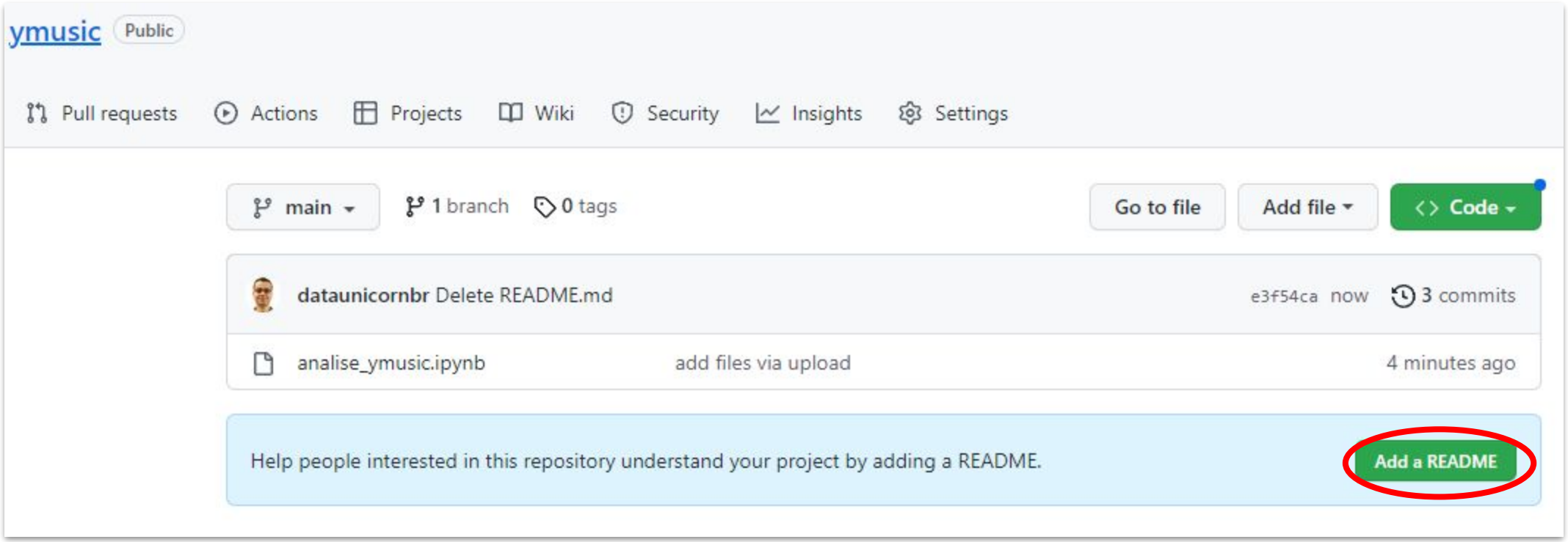
5. Enviar arquivos para repositório criado (1/2)



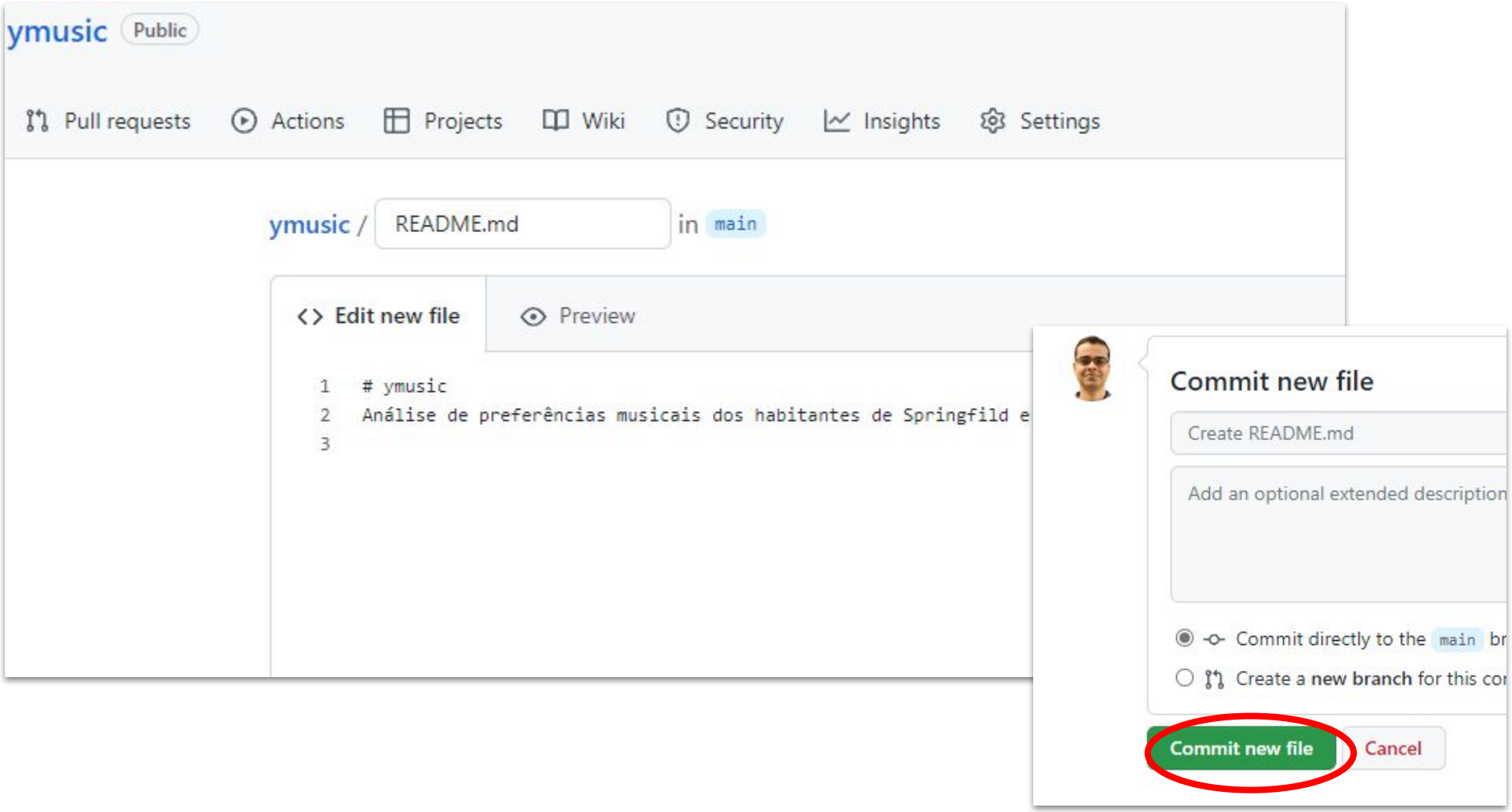
5. Enviar arquivos para repositório criado (2/2)



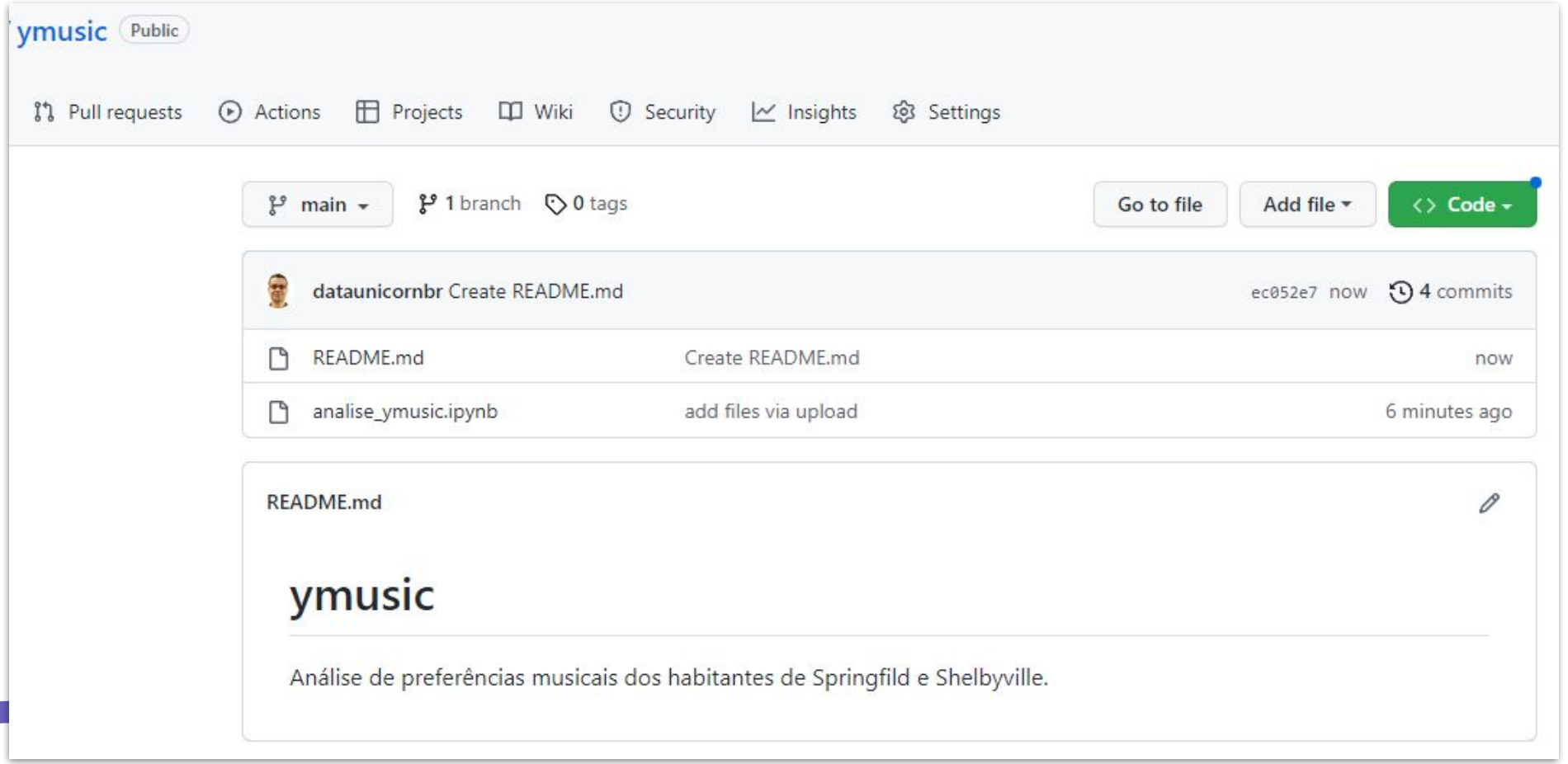
6. Adicionar README ao projeto (1/2)



6. Adicionar README ao projeto (2/2)

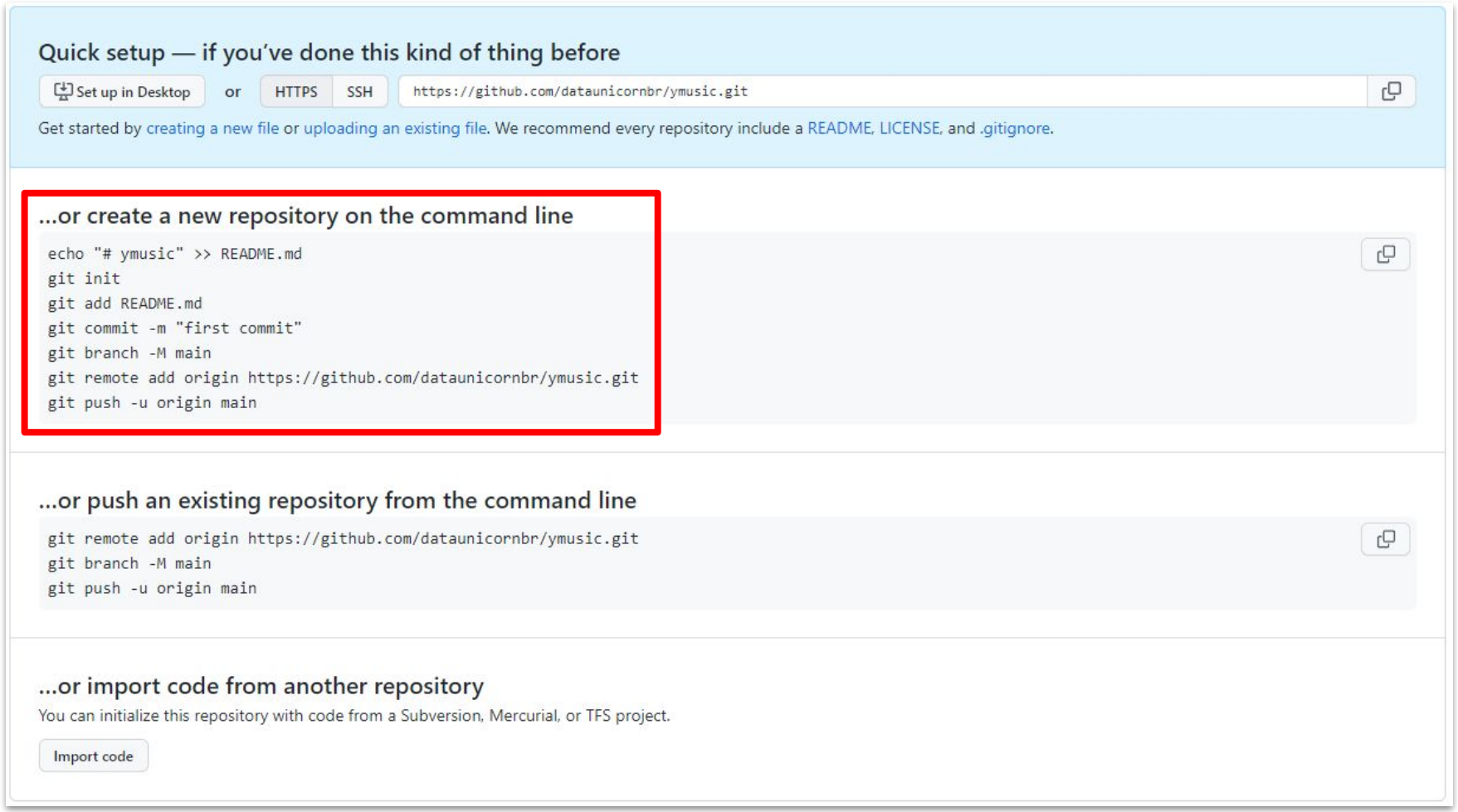


7. Repositório criado e configurado



[dataunicornbr/ymusic: Análise de preferências musicais dos habitantes de Springfild e Shelbyville. (github.com)]

Outras formas... (requer instalação do Git)



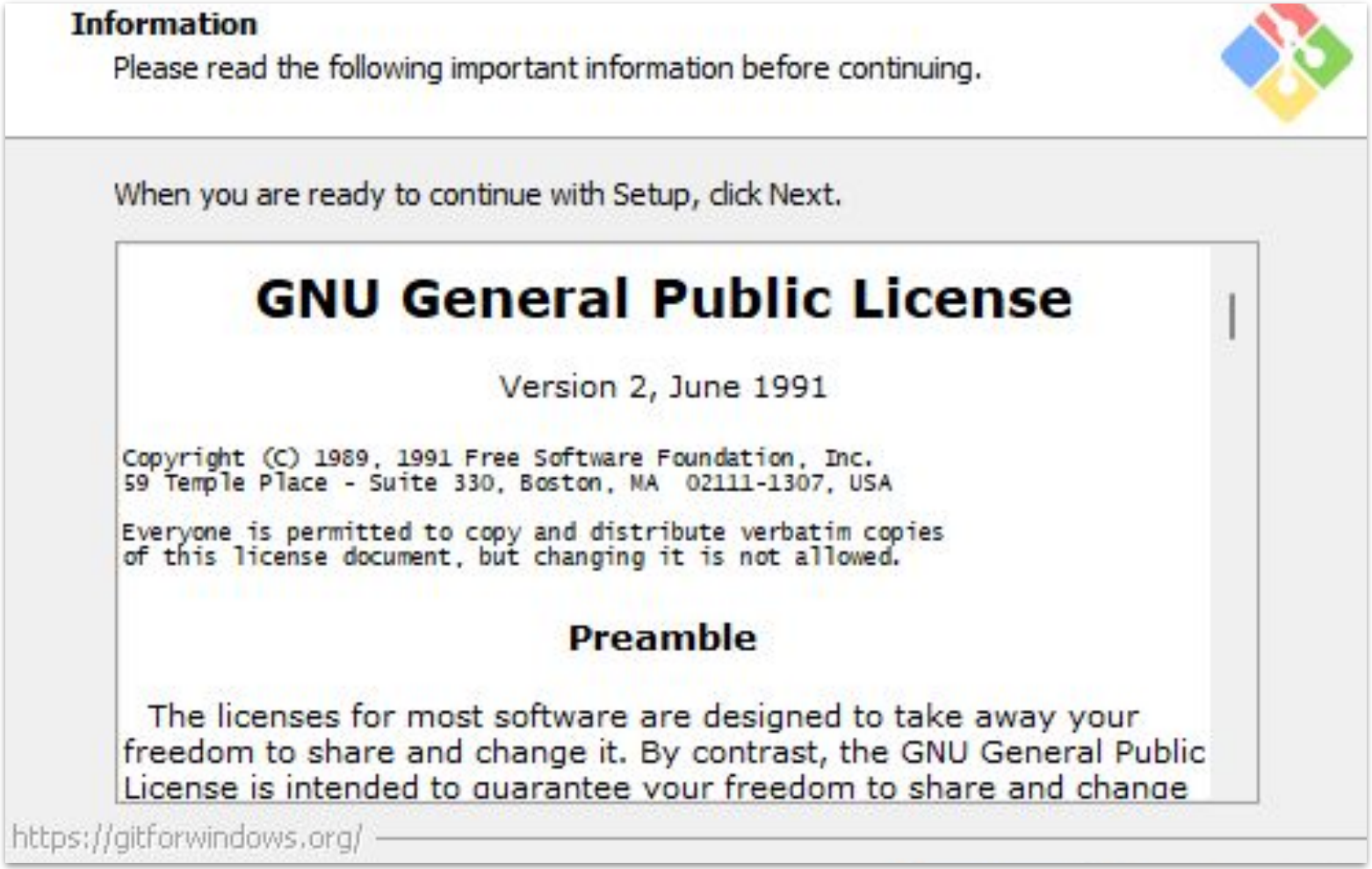
Download do Git

Git (git-scm.com)

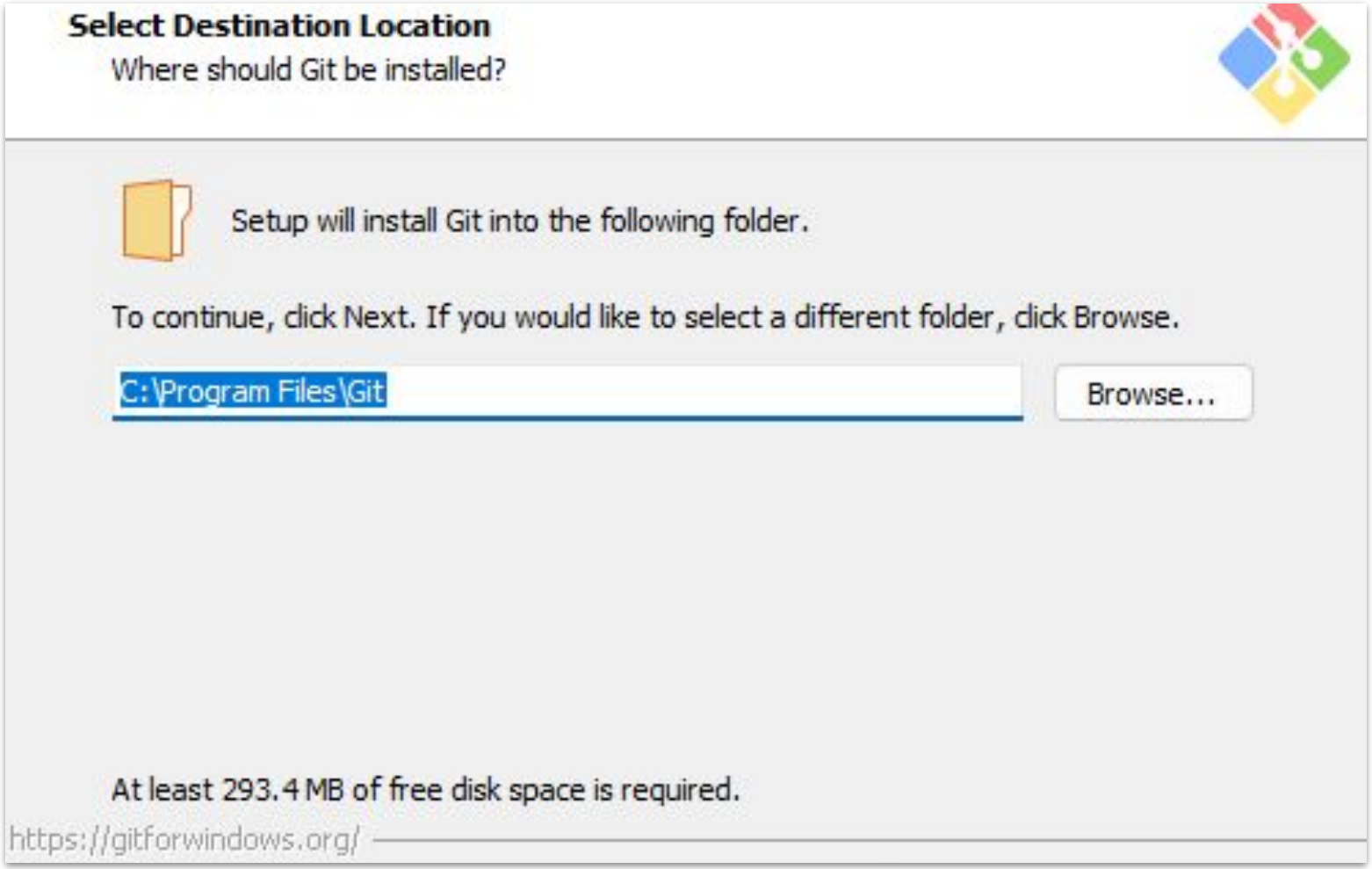
Instalação do Git

Git - Instalando o Git (git-scm.com)

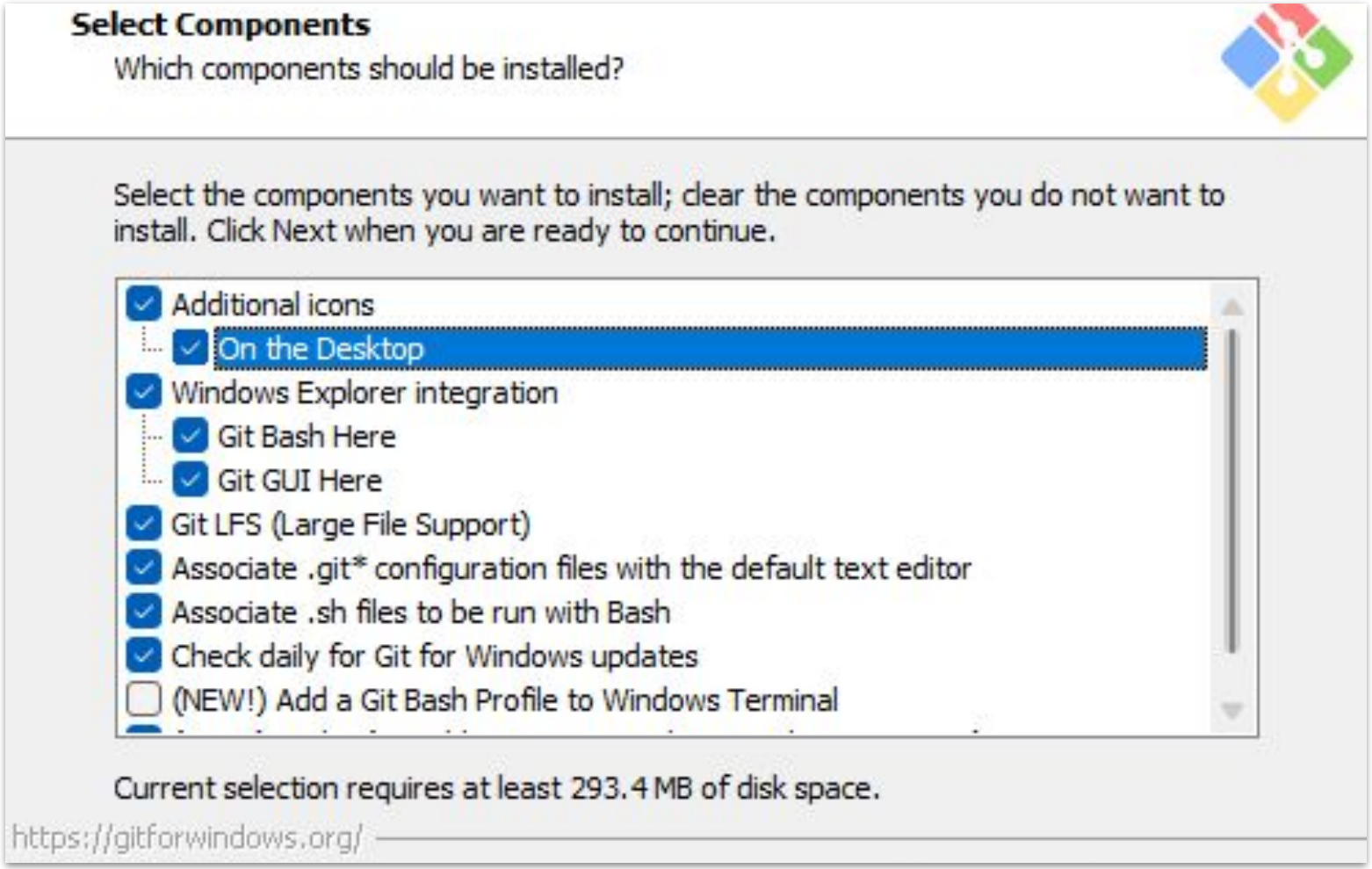
Instalação do Git no Windows



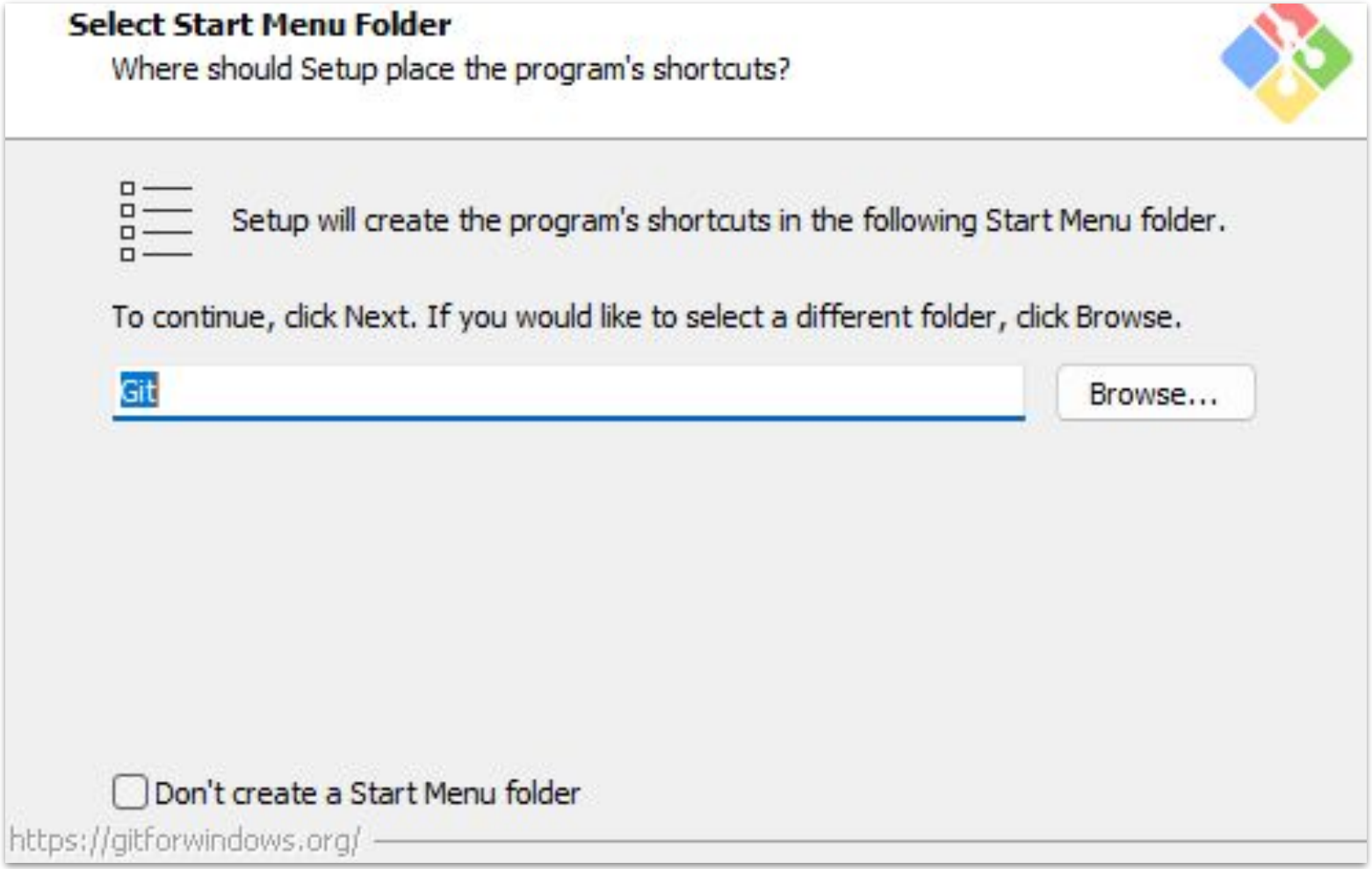
Instalação do Git no Windows



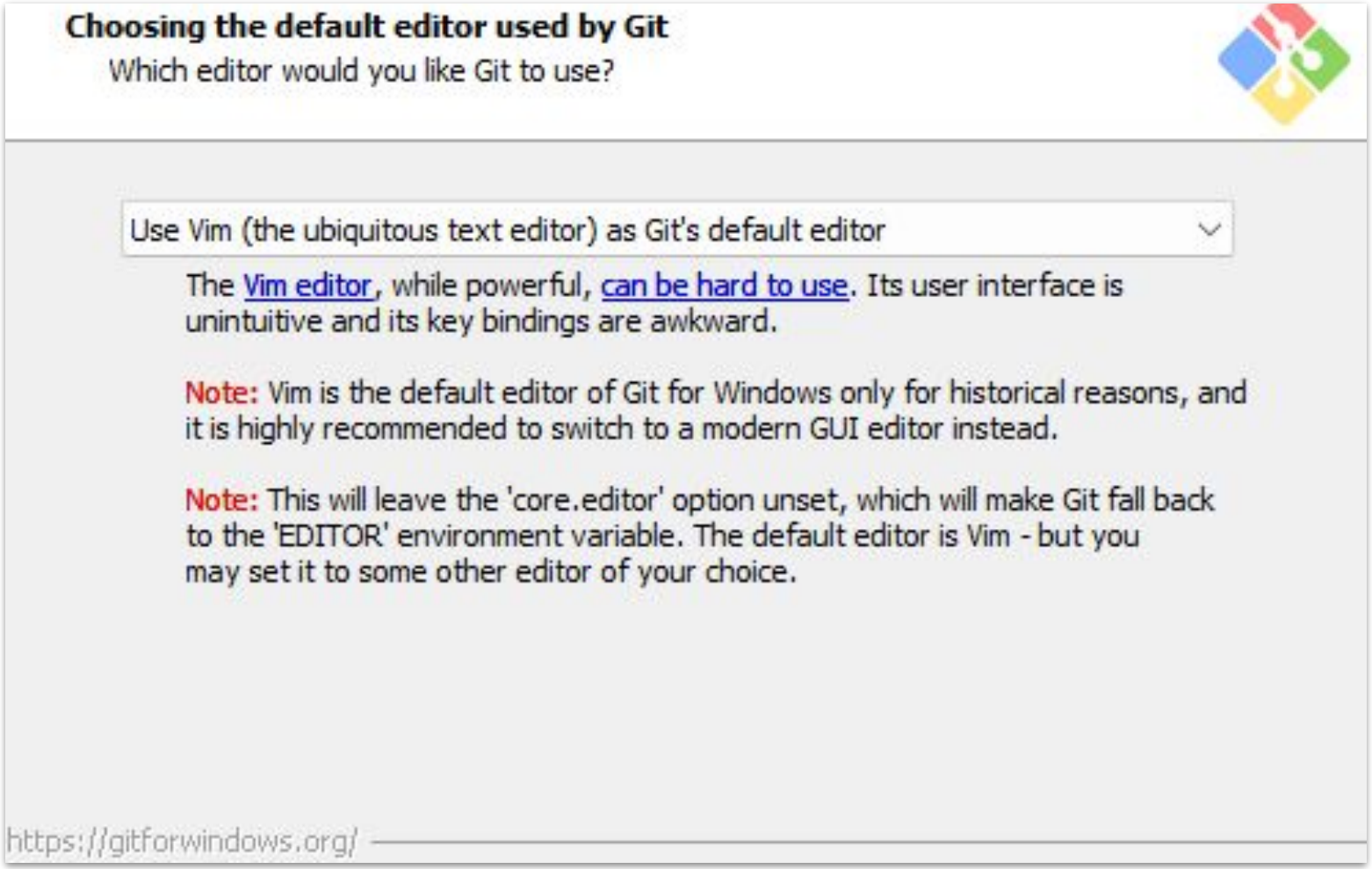
Instalação do Git no Windows



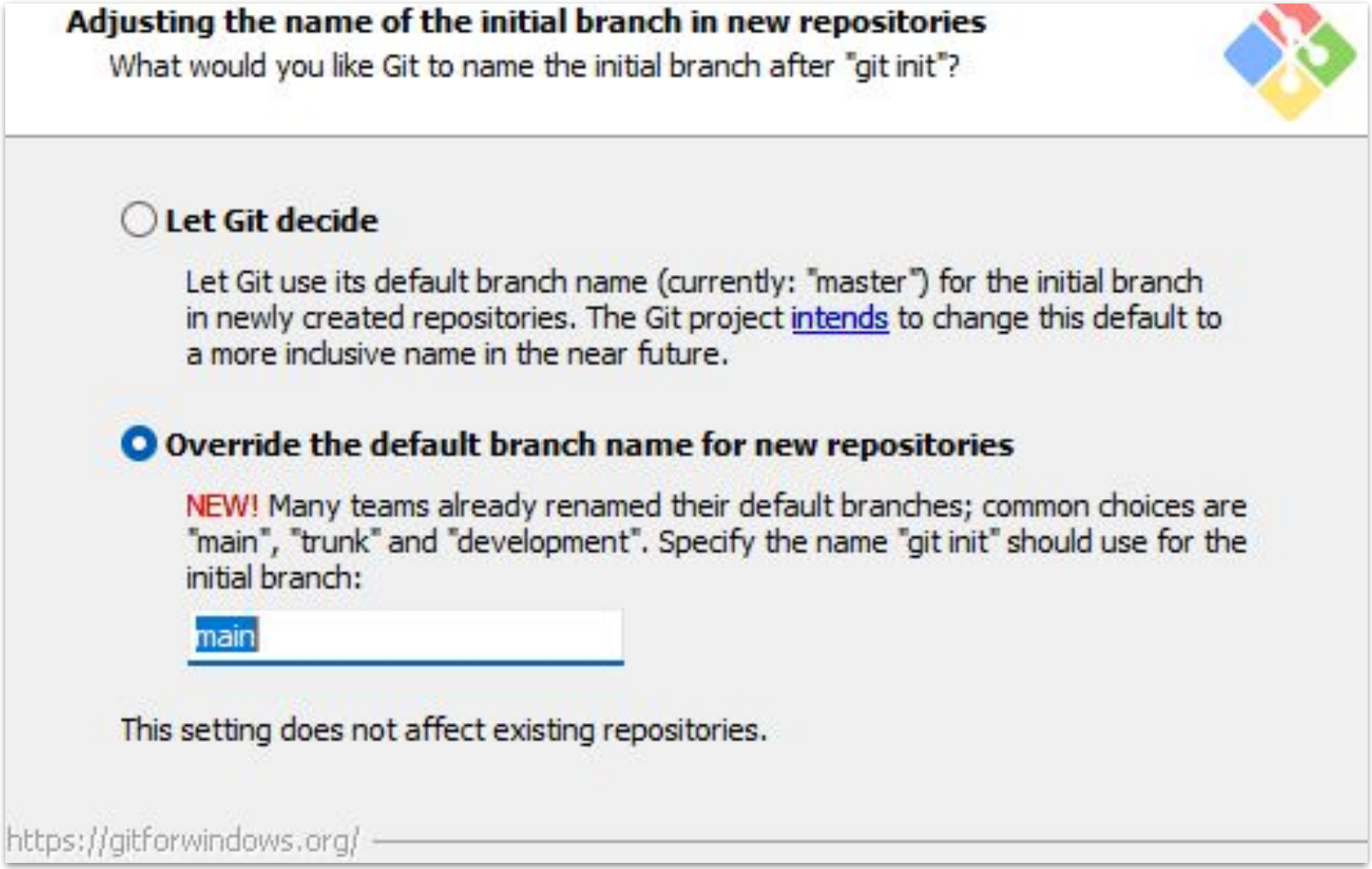
Instalação do Git no Windows



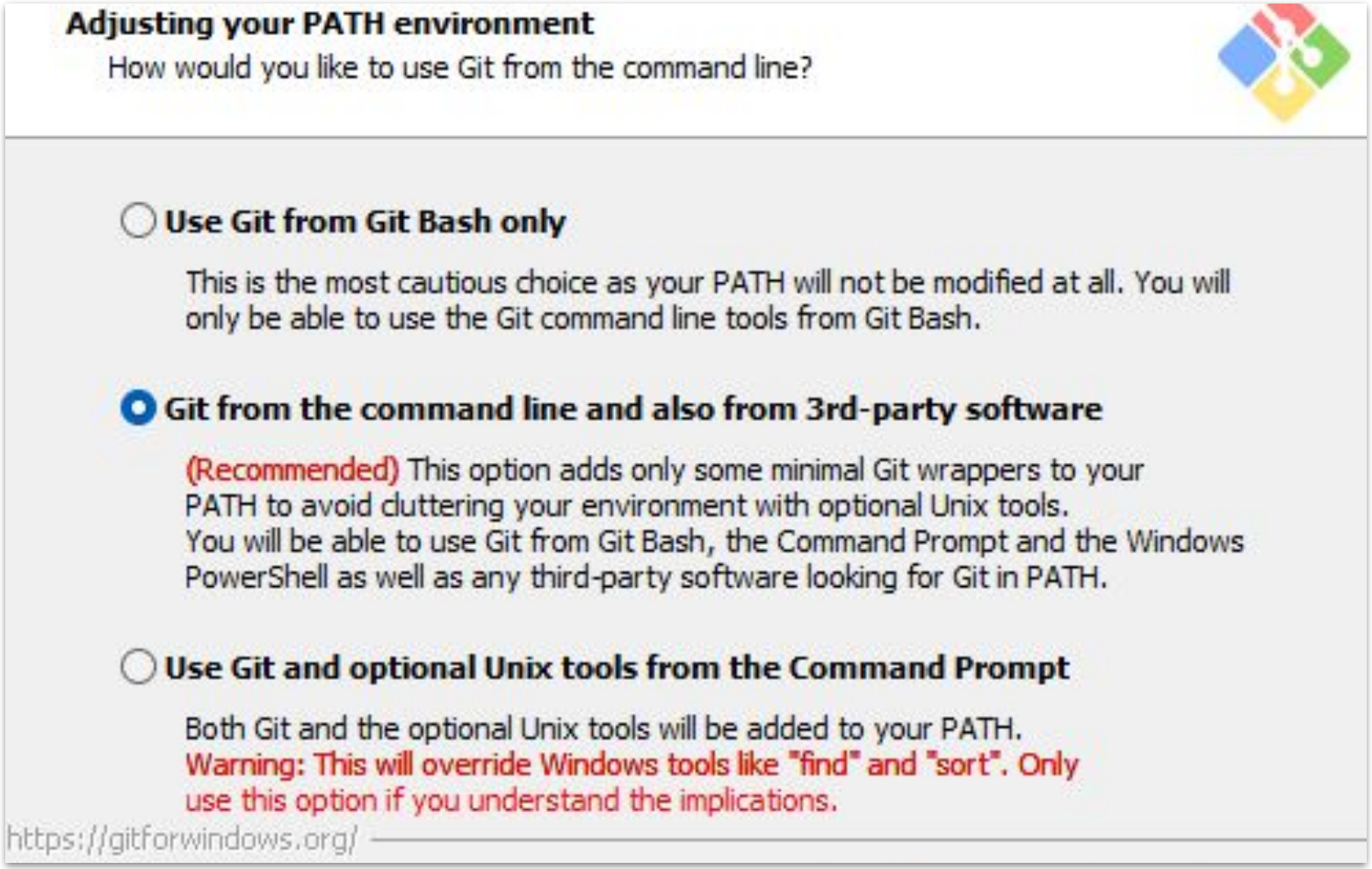
Instalação do Git no Windows



Instalação do Git no Windows

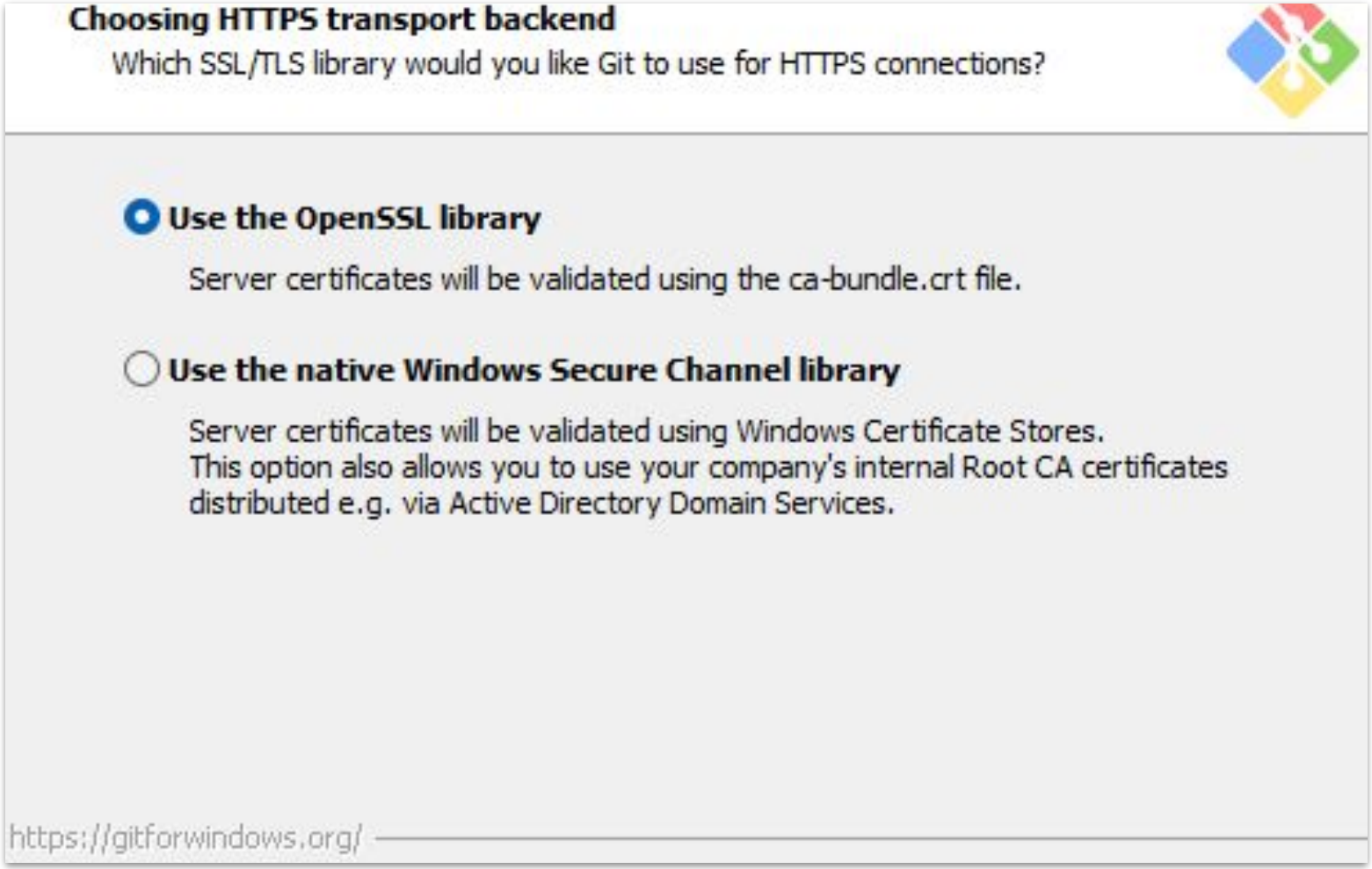


Instalação do Git no Windows

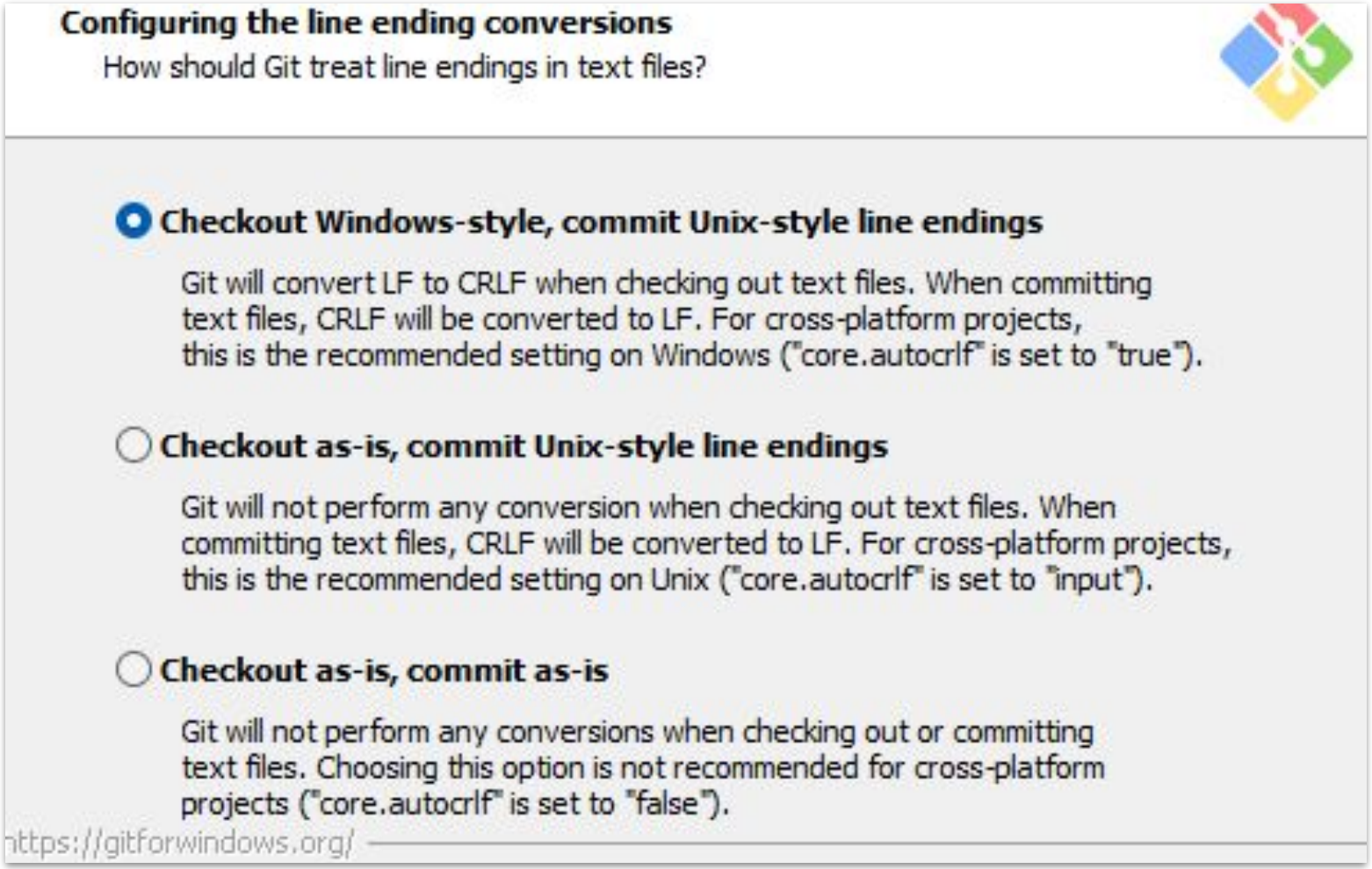


Instalação do Git no Windows

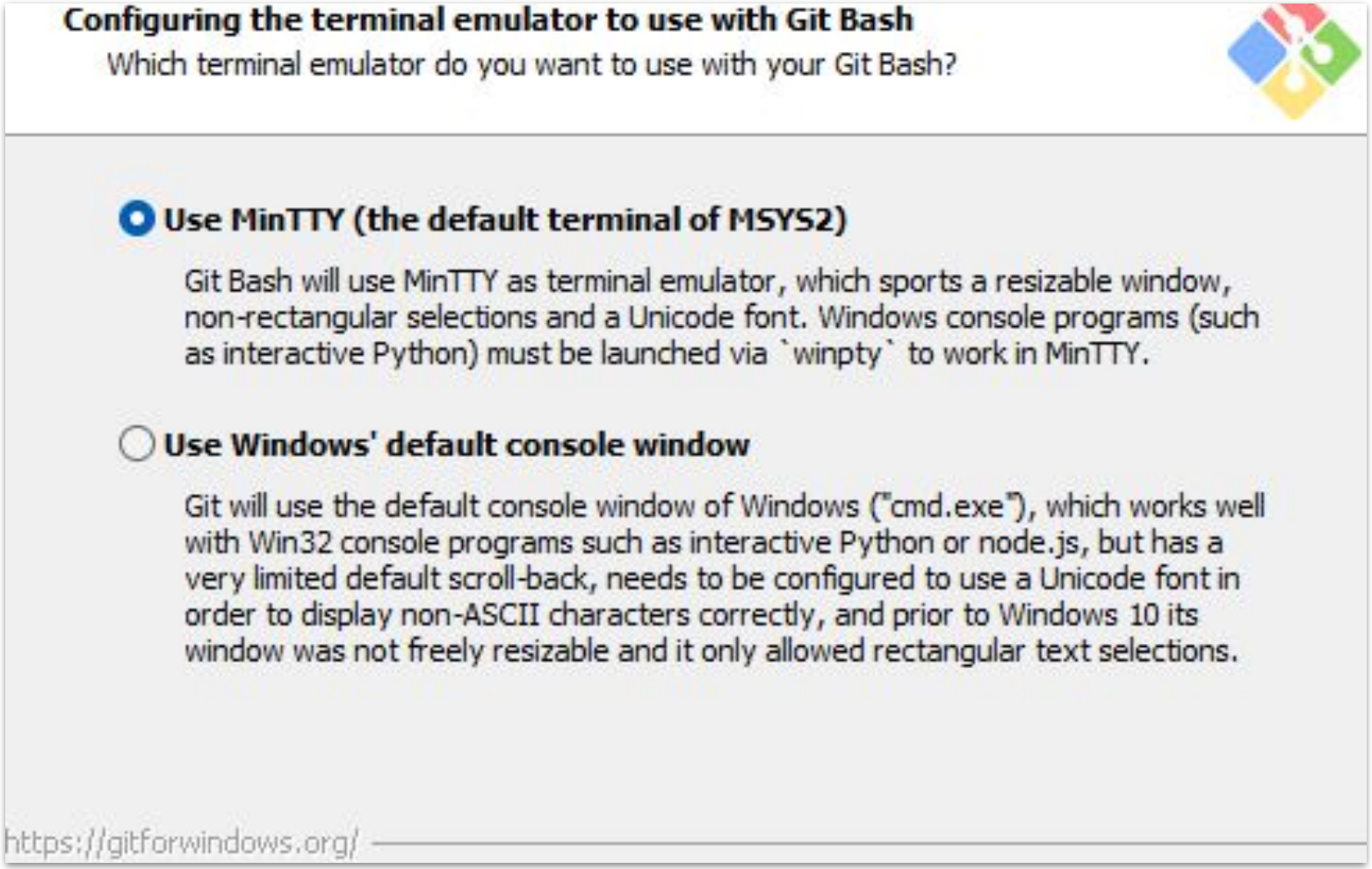
Instalação do Git no Windows



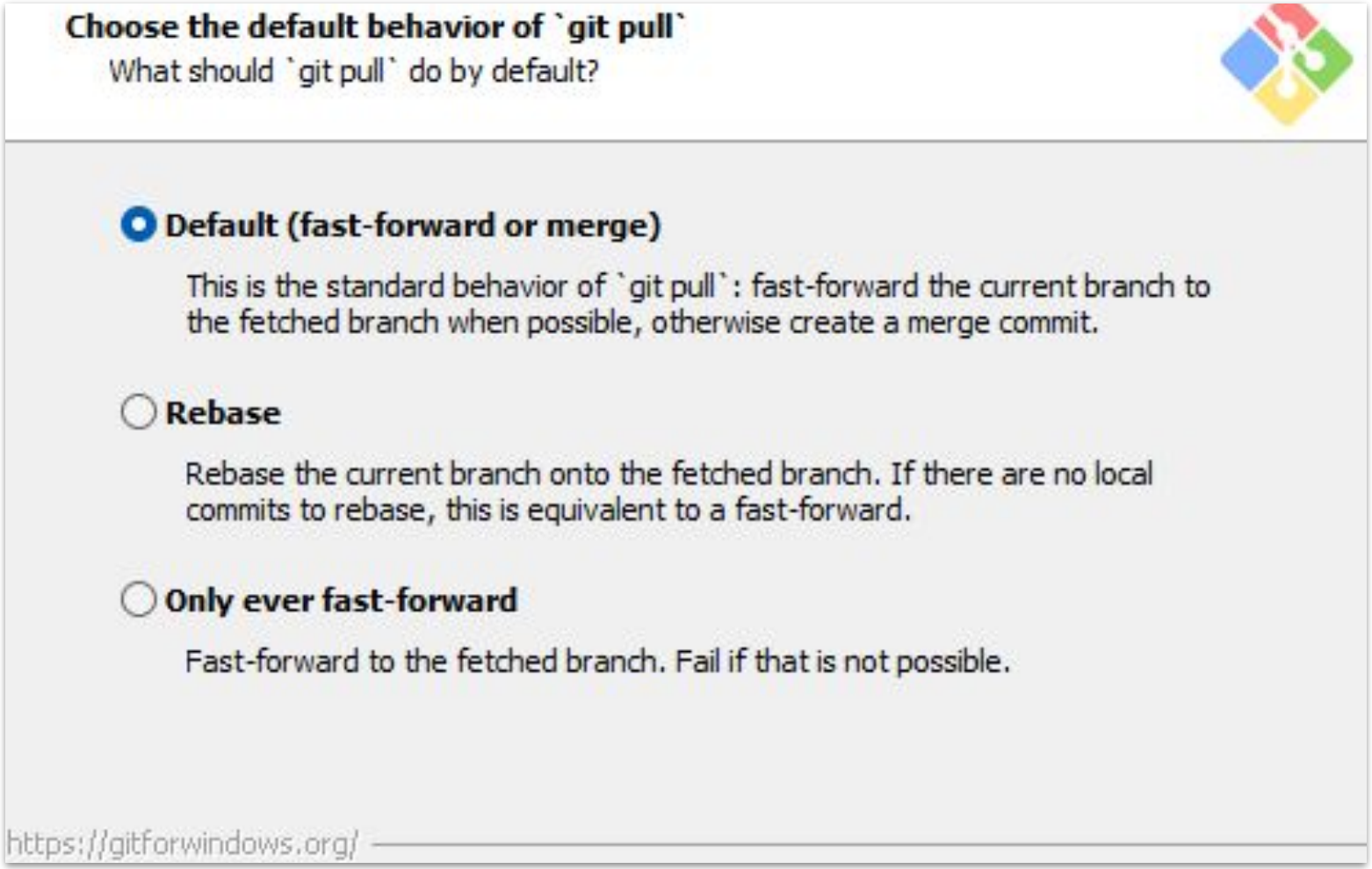
Instalação do Git no Windows



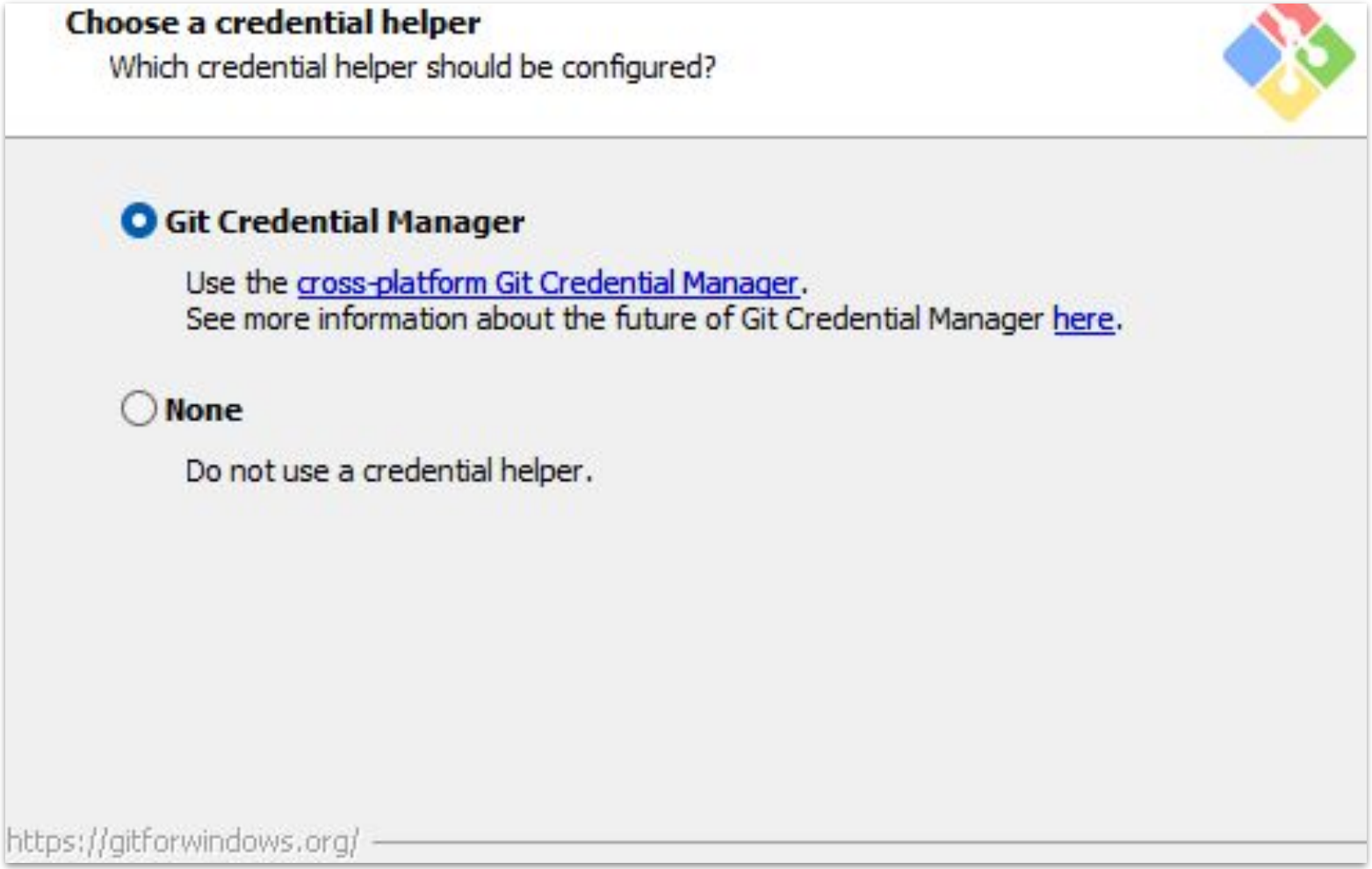
Instalação do Git no Windows



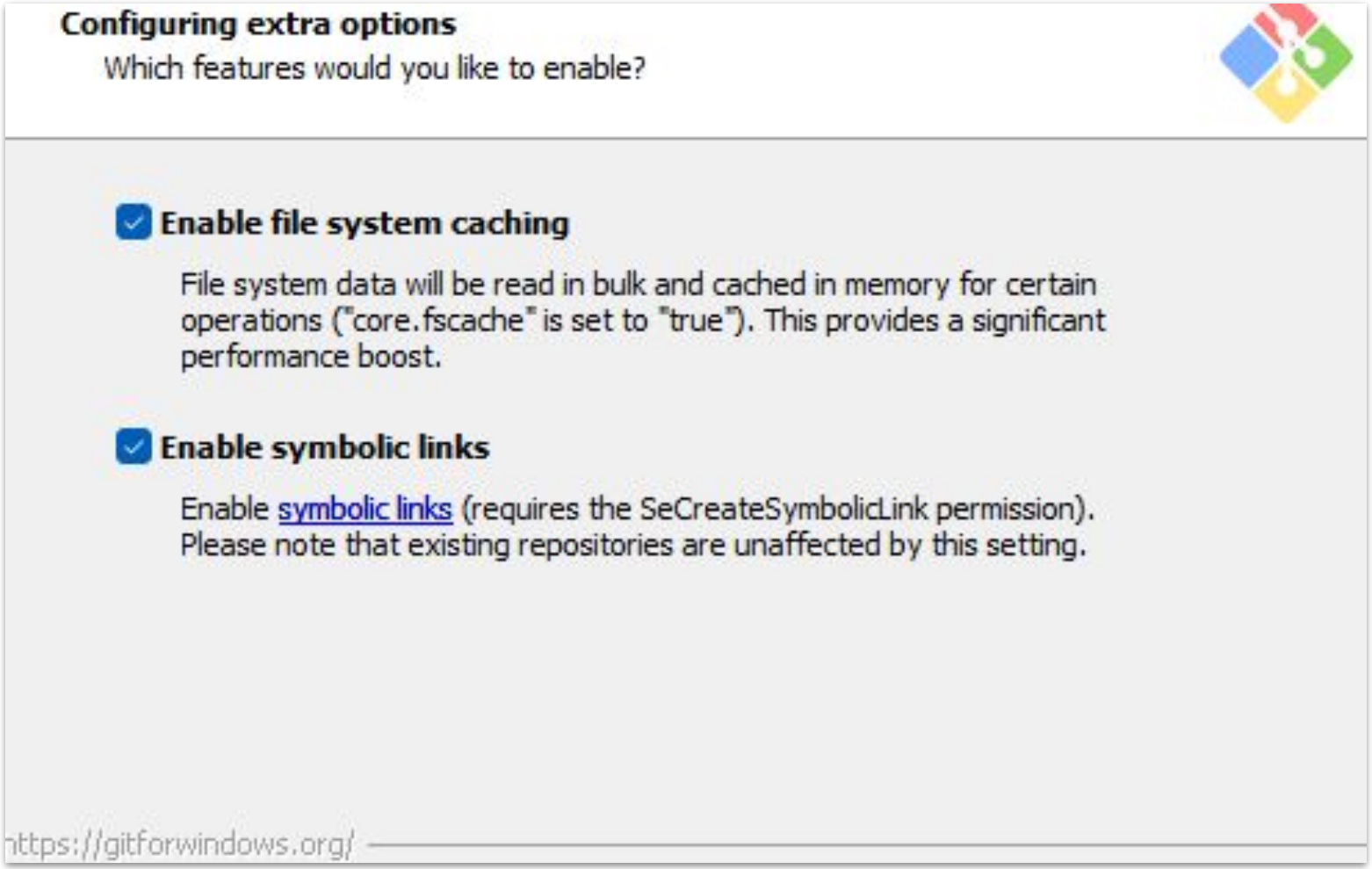
Instalação do Git no Windows



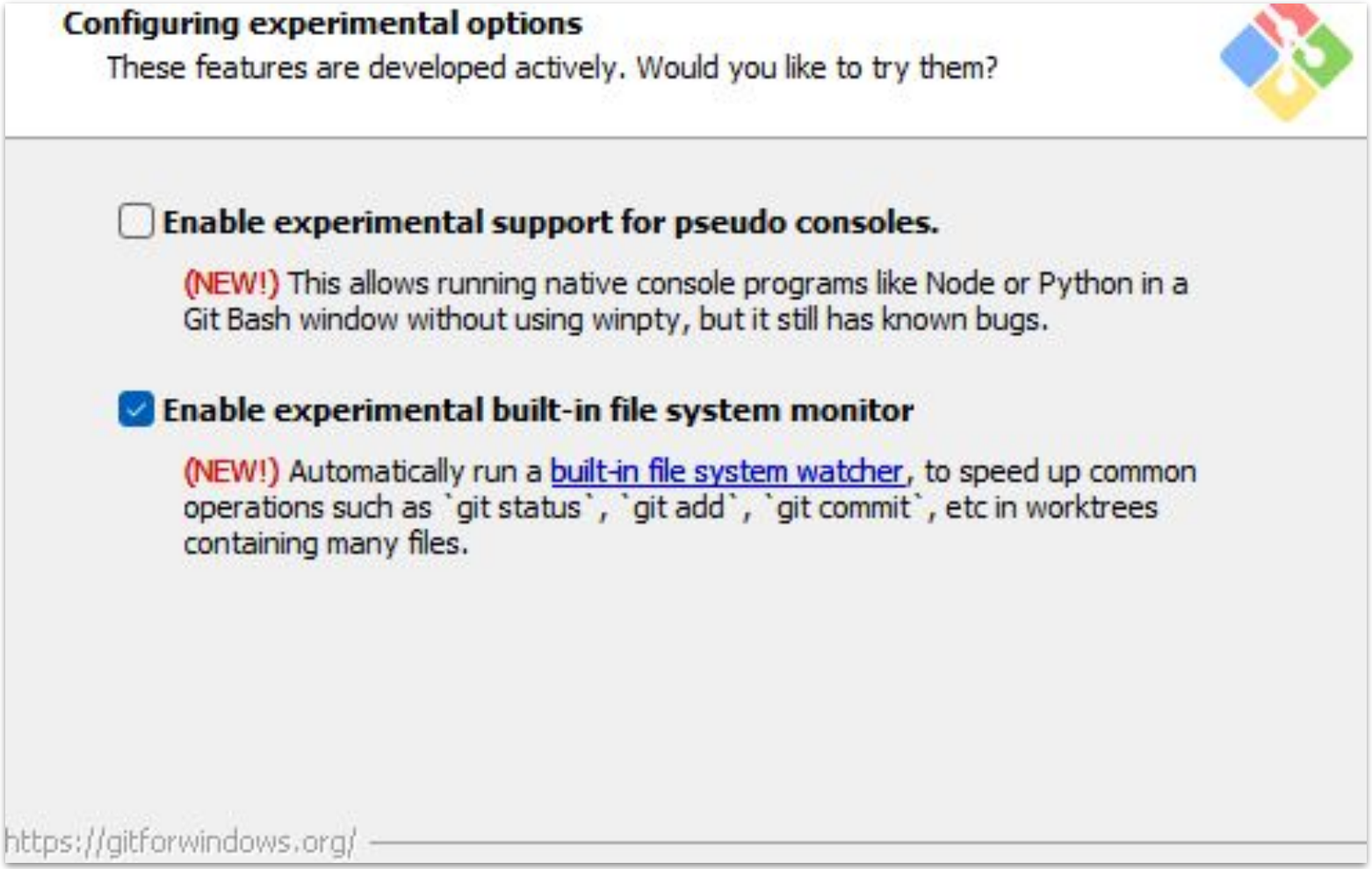
Instalação do Git no Windows



Instalação do Git no Windows

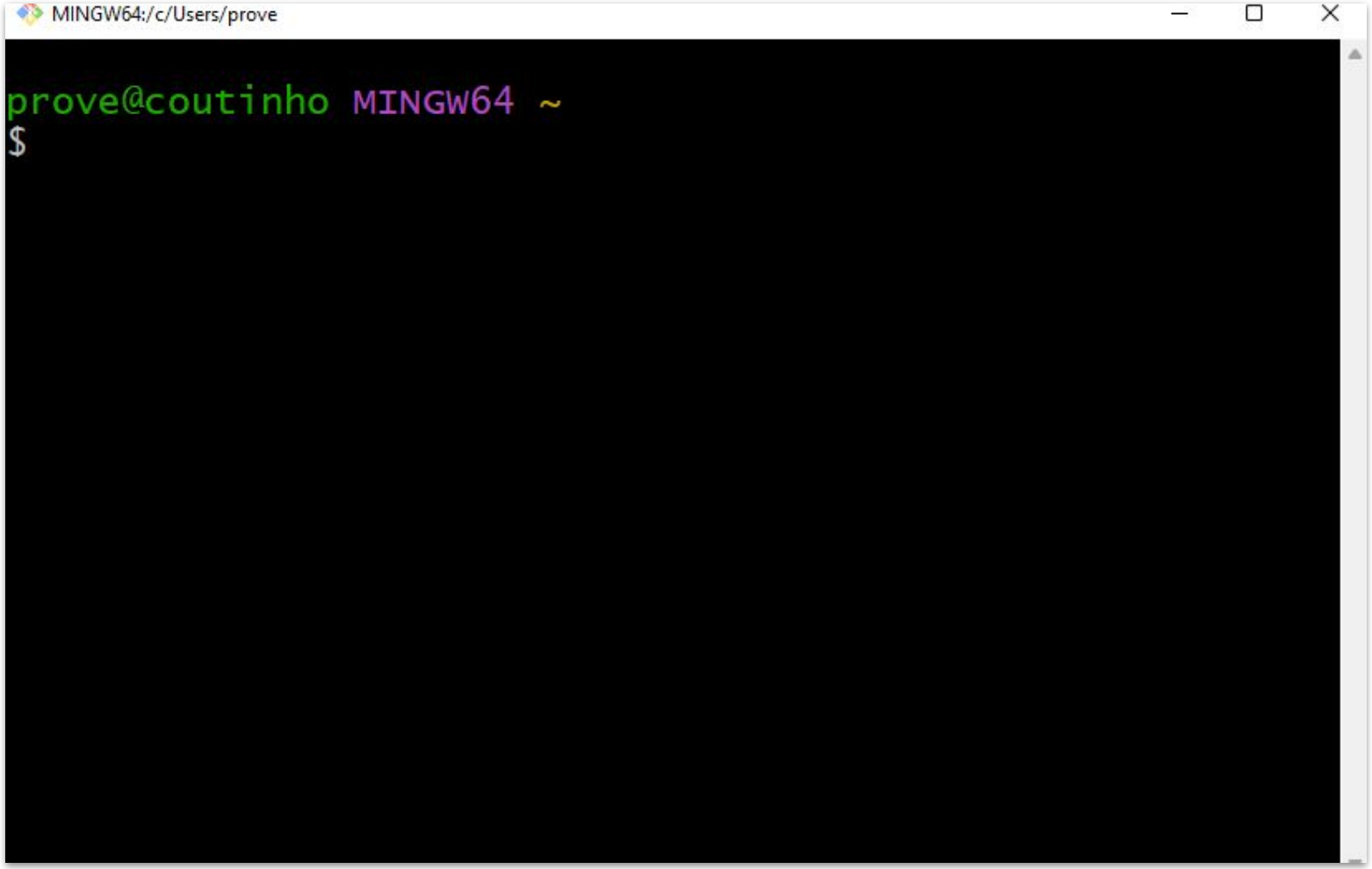


Instalação do Git no Windows



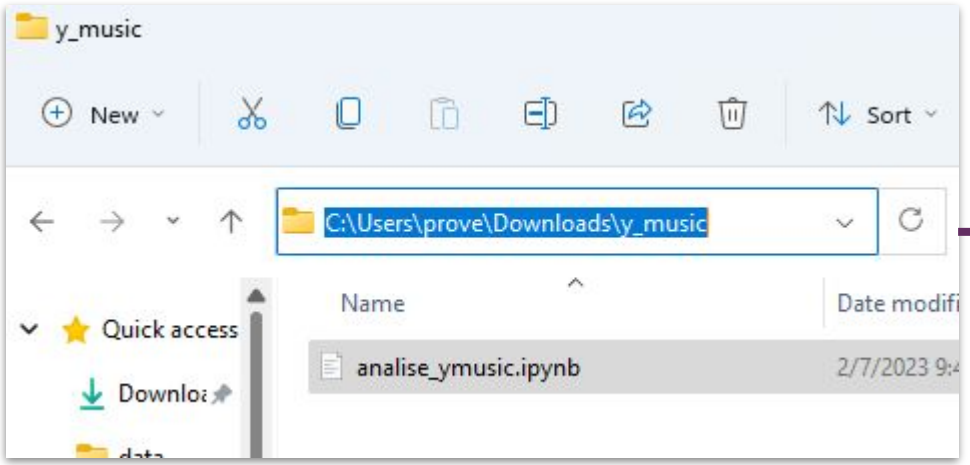
Instalação do Git no Windows

Git Bash



Após instalação do Git, vamos preparar o ambiente

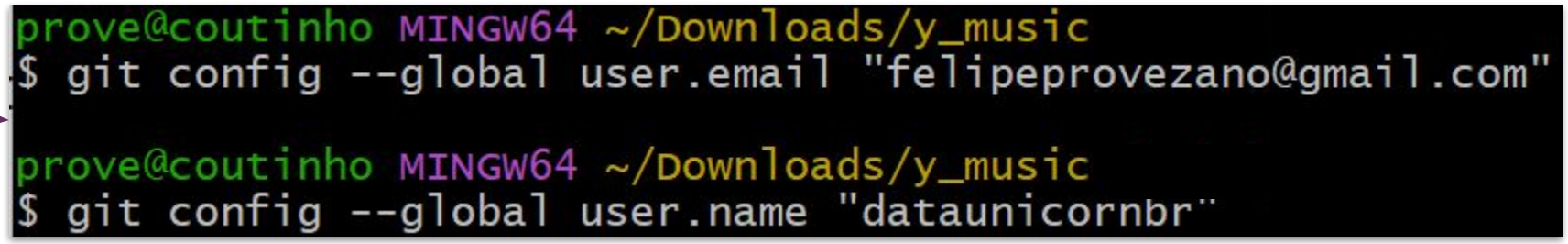
1. Criar pasta com nome do projeto e colocar arquivo notebook dentro da pasta



1. Abrir Git Bash
2. Entrar na pasta do projeto com comando "cd"
3. Verificar se arquivos da pasta do projeto aparecem após comando "ls"

Configurar credenciais do GitHub no Git

git config --global user.email "seu email cadastrado no Github"
git config --global user.name "seu usuário cadastrado no Github"

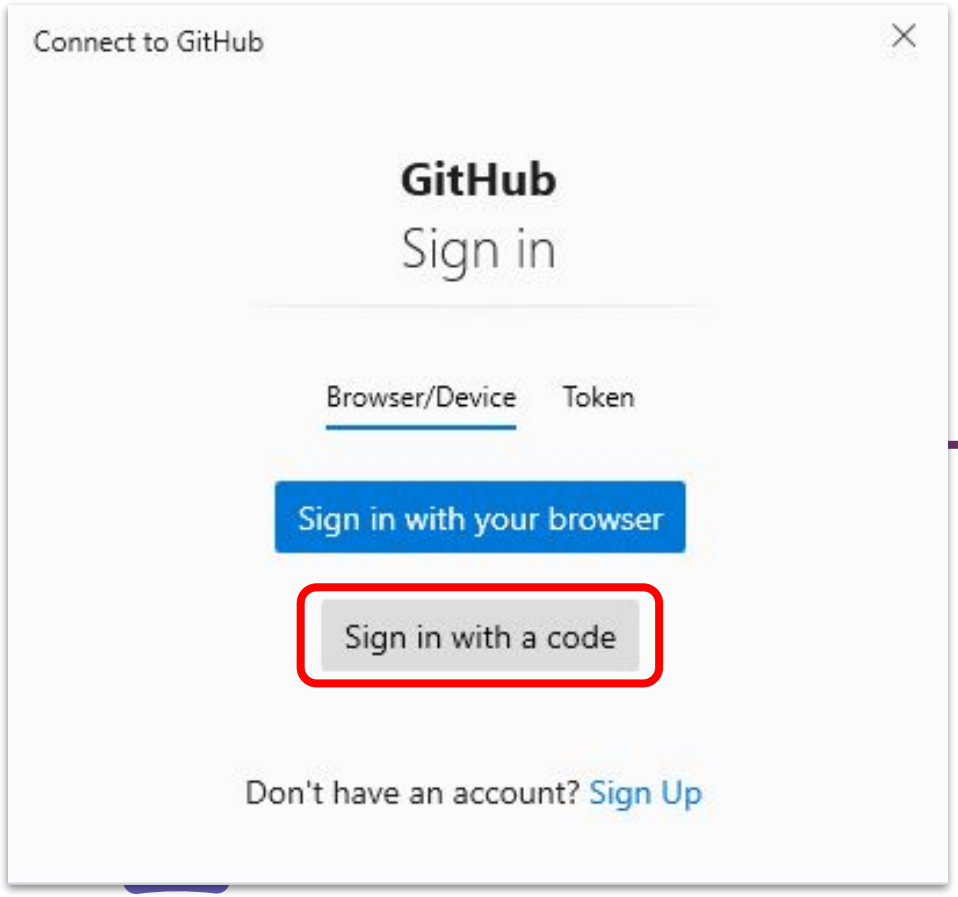


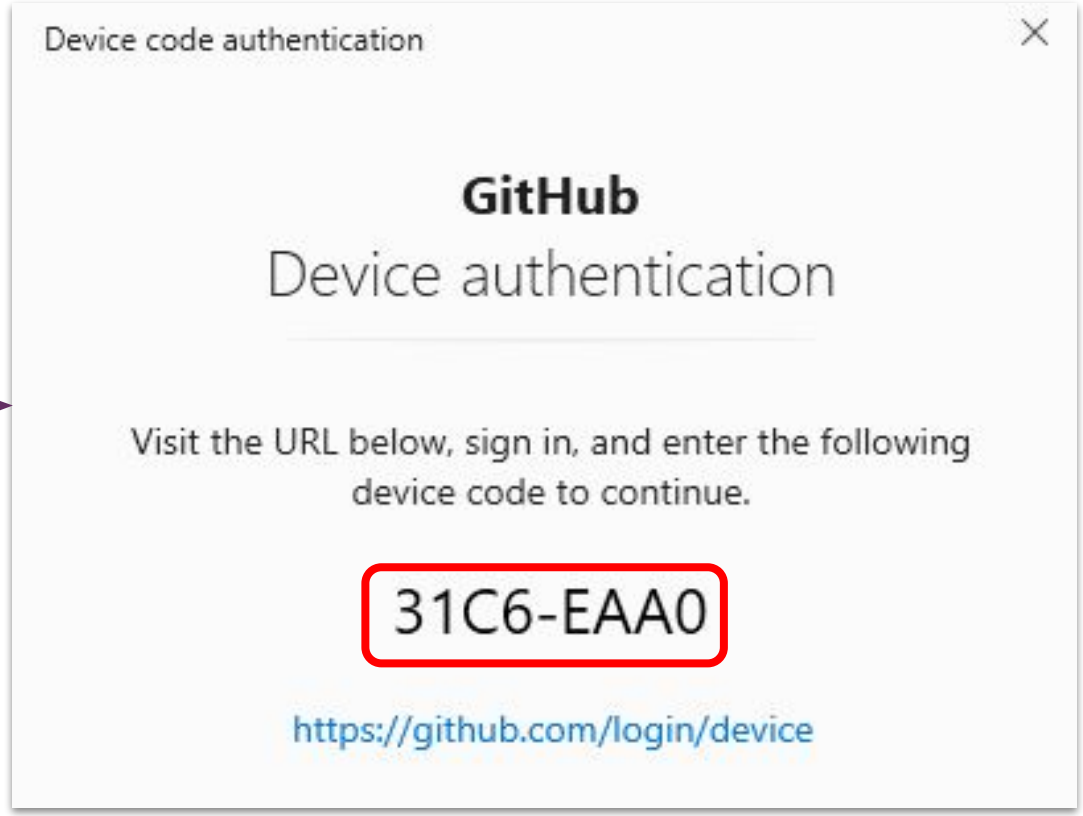
E finalmente, executar os comandos (em ordem)

...or create a new repository on the command line

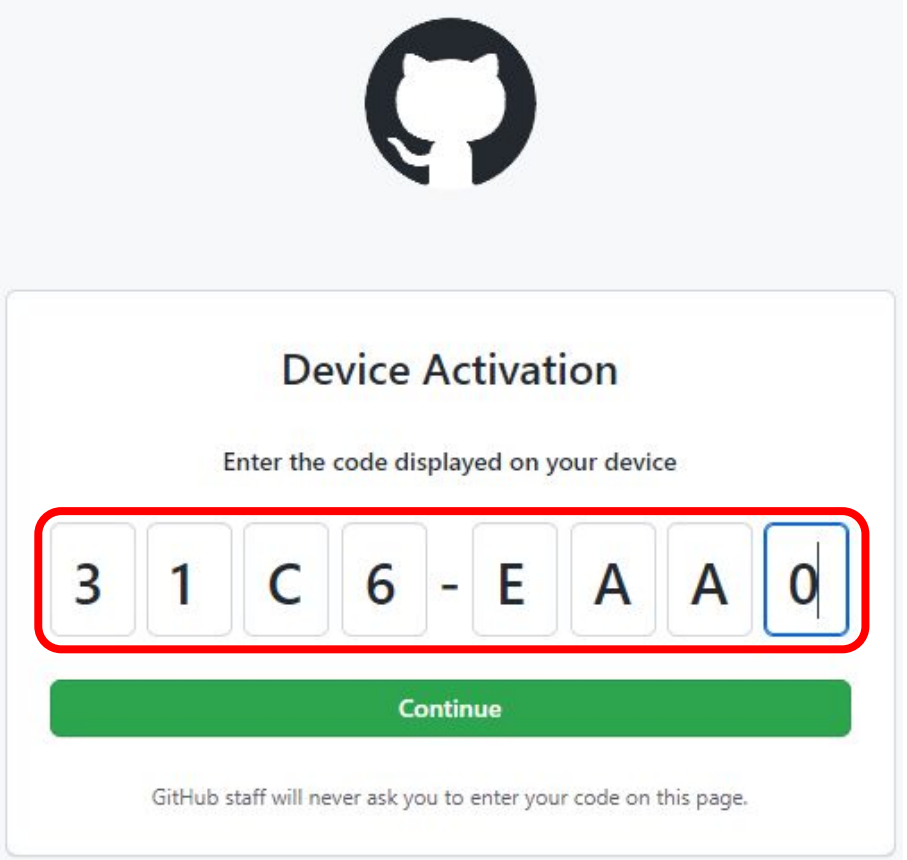
```
echo "# ymusic" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/dataunicornbr/ymusic.git
git push -u origin main
```

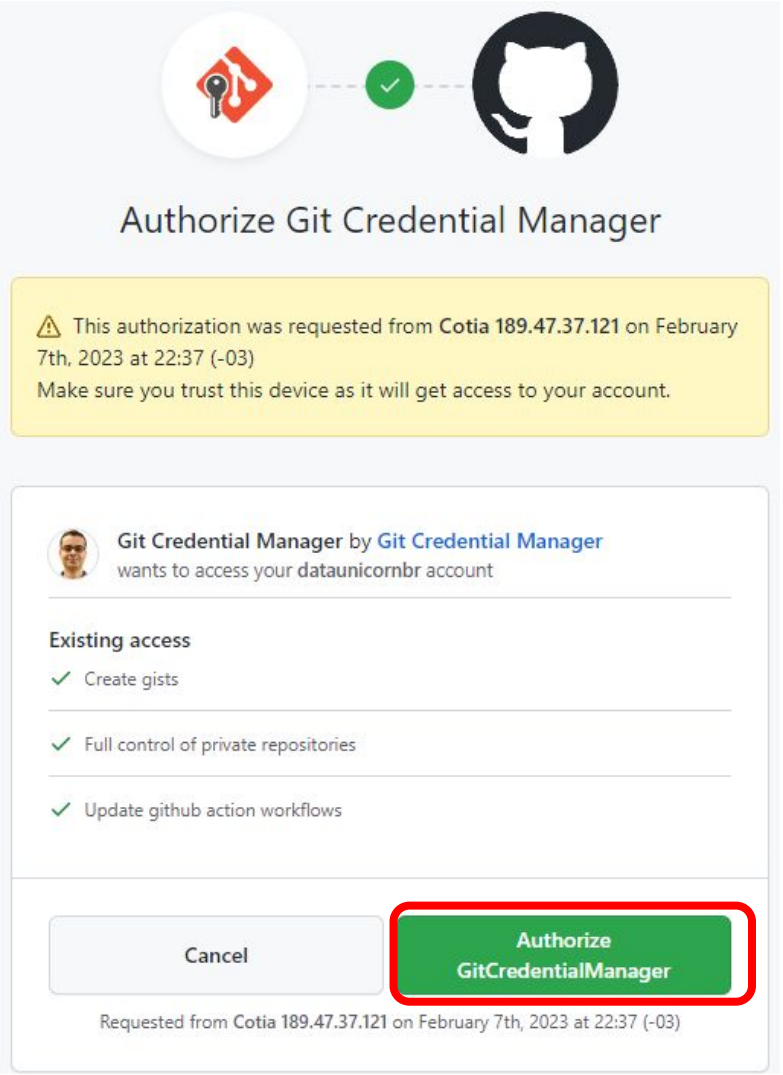
Conectar ao GitHub





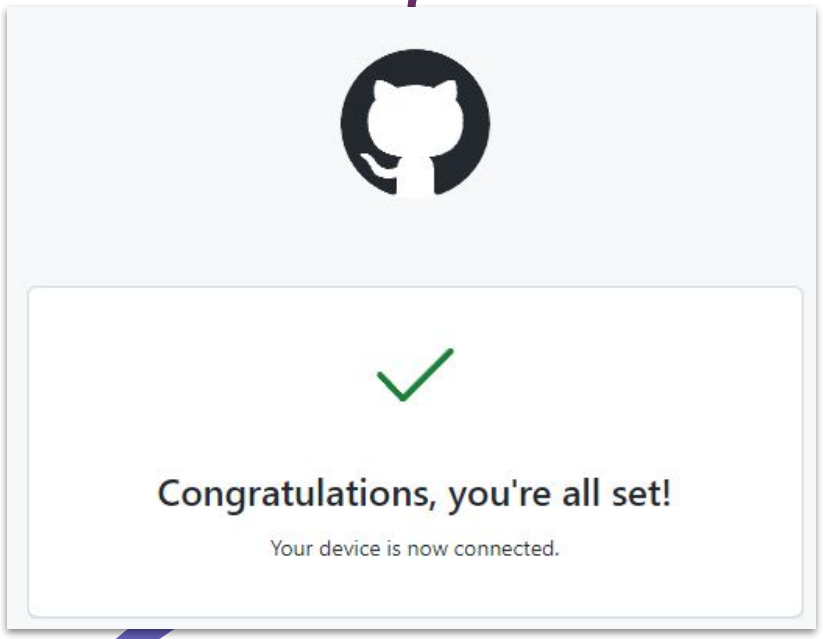
Conectar ao GitHub

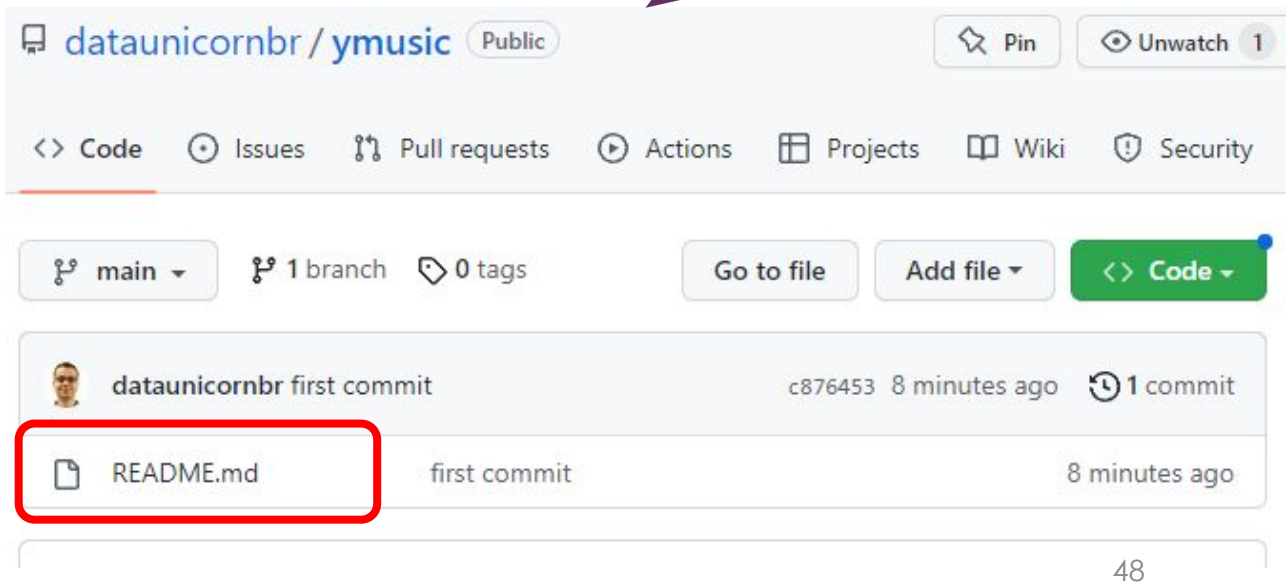




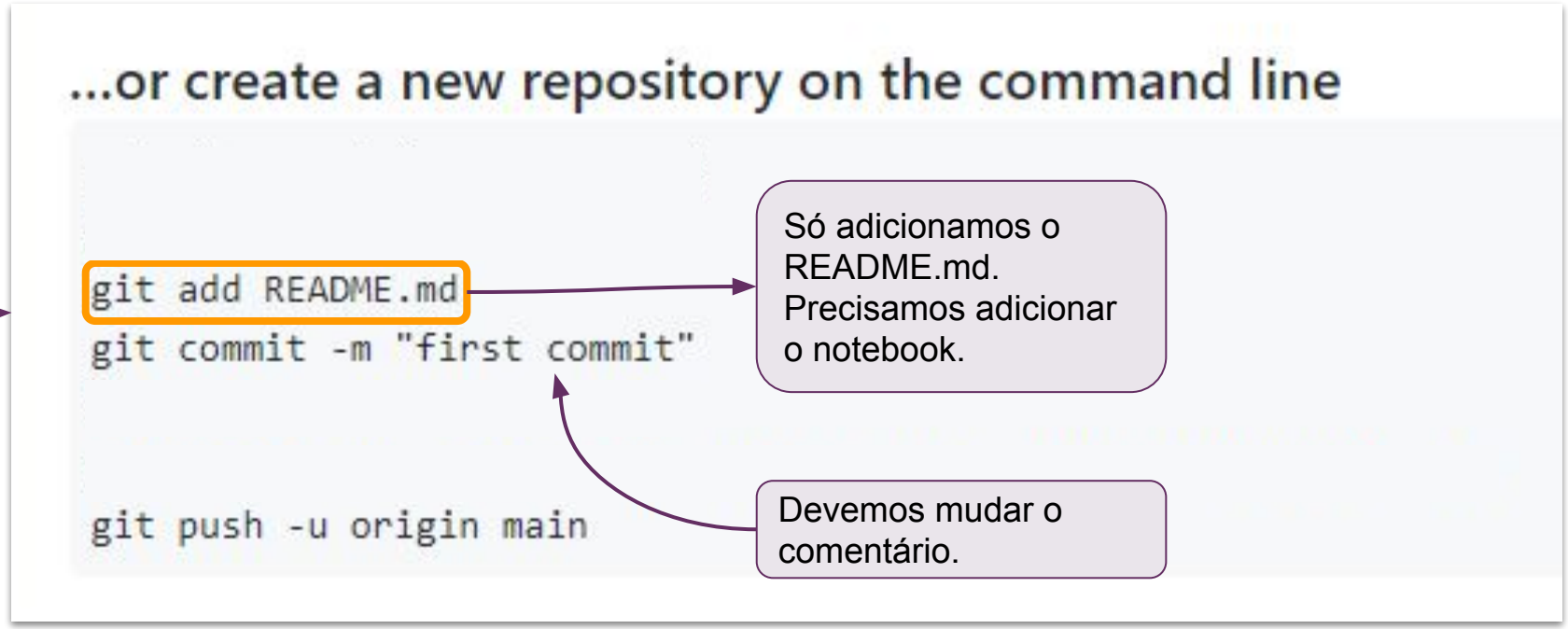
Enviando arquivo “README” ao GitHub

```
$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 226 bytes | 2
Total 3 (delta 0), reused 0 (delta 0), pac
To https://github.com/dataunicornbr/ymusic
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main
```

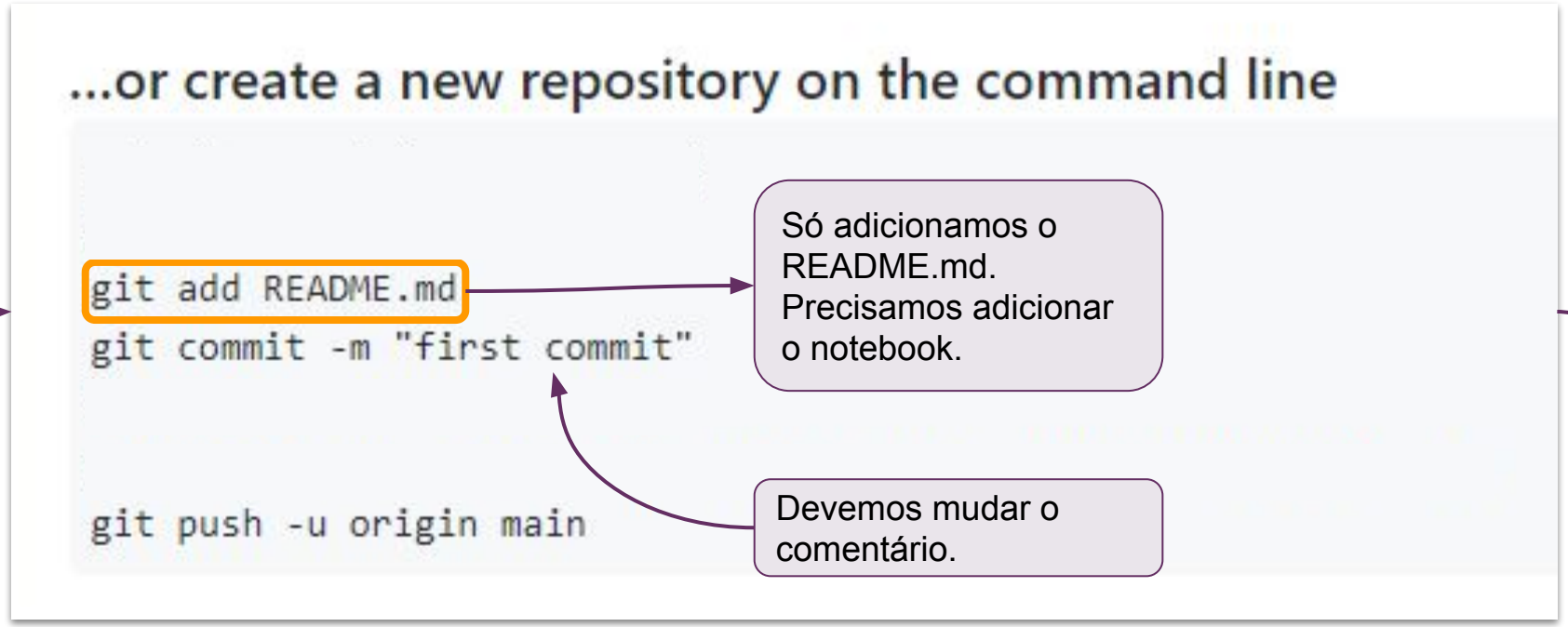




Cadê o notebook (arquivo ipynb)?



Adicionando um novo arquivo



Não precisamos de todos os comandos…

1. git add analise_ymusic.ipynb
2. git commit -m "primeira versão do notebook de análise"
3. git push -u origin main

Sua rotina trabalhando neste projeto

```
prove@coutinho MINGW64 ~/Downloads/y_music    (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

prove@coutinho MINGW64 ~/Downloads/y_music    (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

prove@coutinho MINGW64 ~/Downloads/y_music    (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be co
  (use "git restore <file>..." to discard changes in
        modified:   analise_ymusic.ipynb

no changes added to commit (use "git add" and/or "gi
```

```
prove@coutinho MINGW64 ~/Downloads/y_music    (main)
$ git add analise_ymusic.ipynb
warning: in the working copy of 'analise_ymusic.ipyn
by CRLF the next time Git touches it

prove@coutinho MINGW64 ~/Downloads/y_music    (main)
$ git commit -m "mudei o título do notebook"
[main 8ecddd4] mudei o título do notebook
 1 file changed, 1 insertion(+), 1 deletion(-)

prove@coutinho MINGW64 ~/Downloads/y_music    (main)
$ git push -u origin main
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 353 bytes | 353.00 KiB/
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with
To https://github.com/dataunicornbr/ymusic.git
   fb0a3a5..8ecddd4  main -> main
branch 'main' set up to track 'origin/main'.
```

E se eu quiser me aprofundar em Git e GitHub?

Documentation

Reference

Reference Manual

The official and comprehensive **man pages** that are included in the Git package itself.

Quick reference guides: **GitHub Cheat Sheet** | **Visual Git Cheat Sheet**

Book

Pro Git

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.

Videos

What is Version Control?
Length: 05:59

What is Git?
Length: 08:15

Get Going with Git
Length: 04:26

Quick Wins with Git
Length: 05:06

See all videos →

Git - Documentation (git-scm.com)

Git e GitHub

Felipe P. Coutinho