

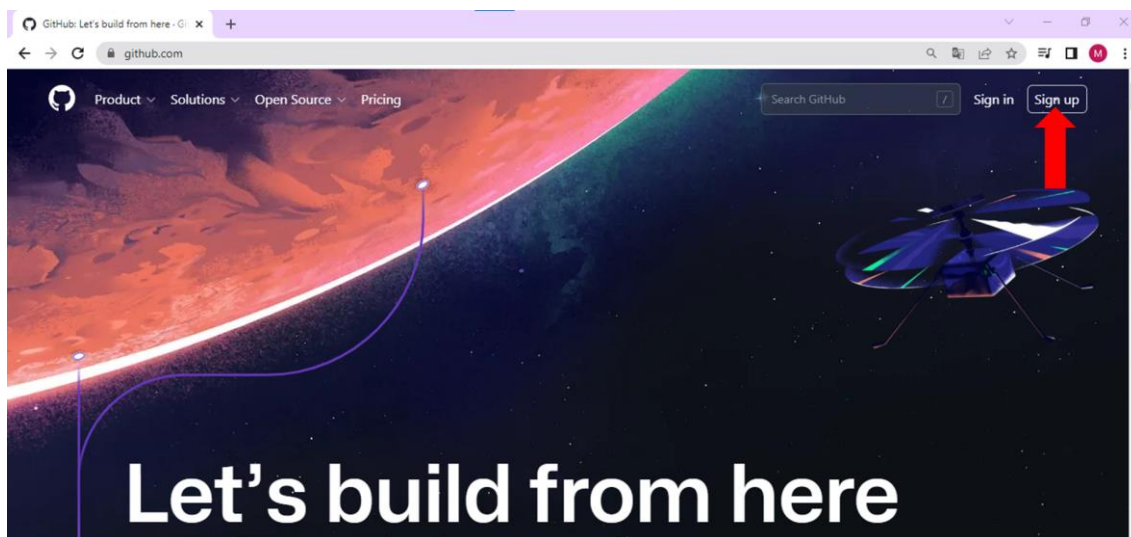
Manual de Instalação do Git

Instalação do GitBash e GitHub Desktop para Windows

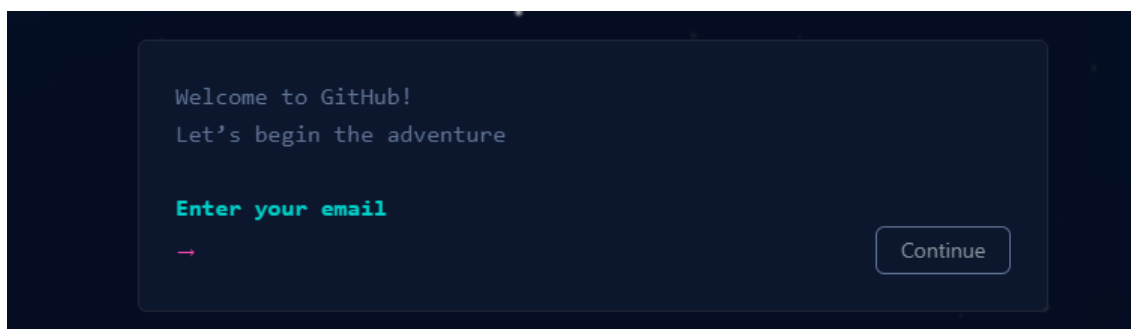
Por Milene Moraes

1- Criar conta:

A princípio é necessário criar uma conta no GitHub (caso já tenha pular para o tópico 2), para isso é só acessar <http://www.github.com/> e selecionar Sign up (em português inscrever-se) no canto superior direito (com uma seta vermelha na imagem na sequência).



Na sequência é necessário inserir o e-mail para conta, conforme imagem abaixo (Dica: Utilize seu e-mail pessoal principal).



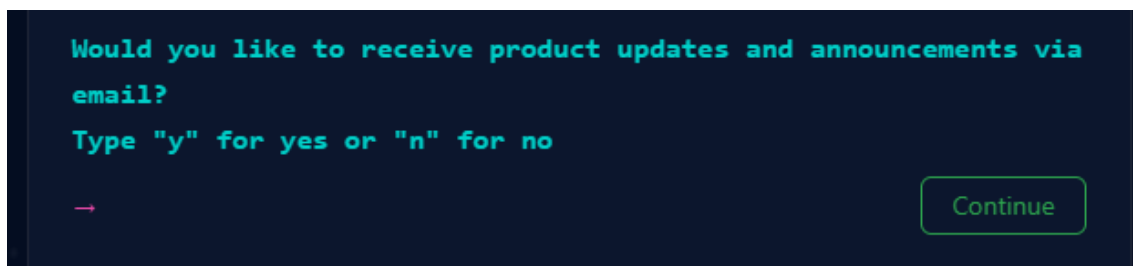
Também é requerida a criação de uma senha, conforme imagem.



E um nome de usuário que não pode ter o caractere espaço (não pode ser como "Nome Sobrenome", devido ao espaço).



É questionado se você gostaria de receber atualizações de produtos e anúncios por e-mail, caso sim deve ser respondido “y” e caso não “n”.



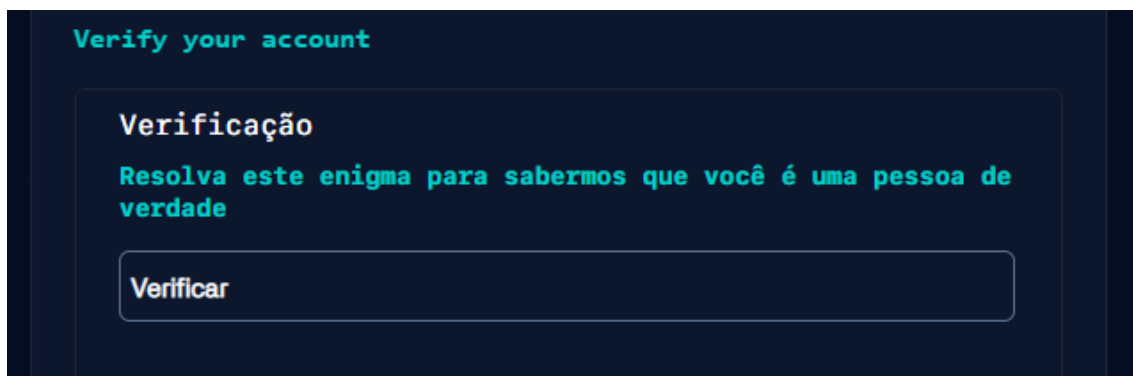
Would you like to receive product updates and announcements via email?

Type "y" for yes or "n" for no

→

Continue

Além disso é necessário realizar a verificação da conta para garantir que não é um robô que está criando a mesma.



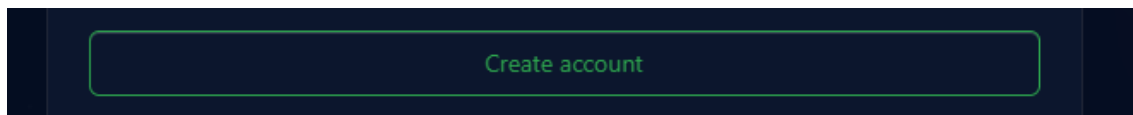
Verify your account

Verificação

Resolva este enigma para sabermos que você é uma pessoa de verdade

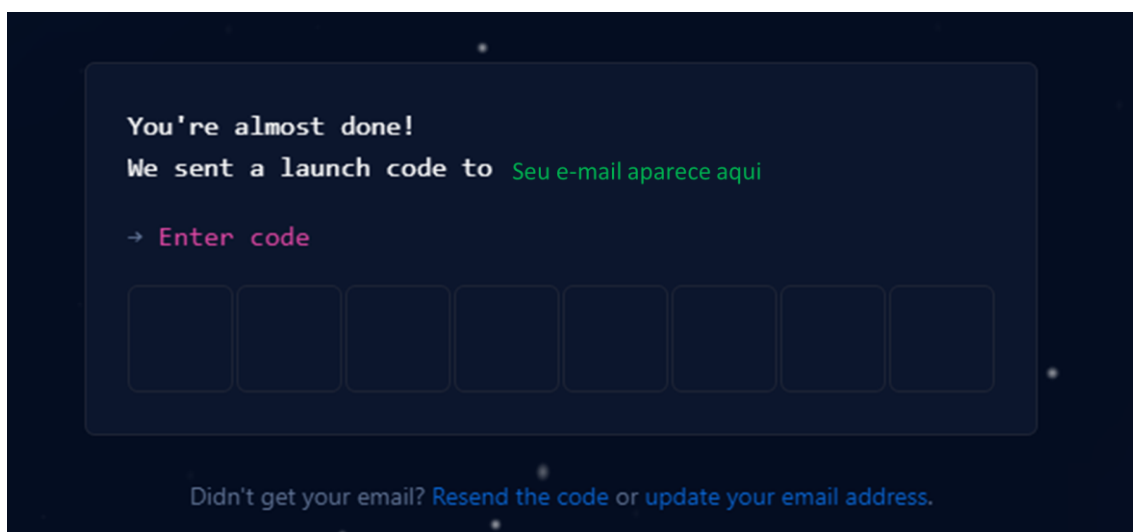
Verificar

Feito isso é necessário selecionar “Create account” na sequência do formulário.



Create account

Com isso você receberá um e-mail com um código que deverá ser utilizado no site para confirmar seu endereço de e-mail.



You're almost done!

We sent a launch code to [Seu e-mail aparece aqui](#)

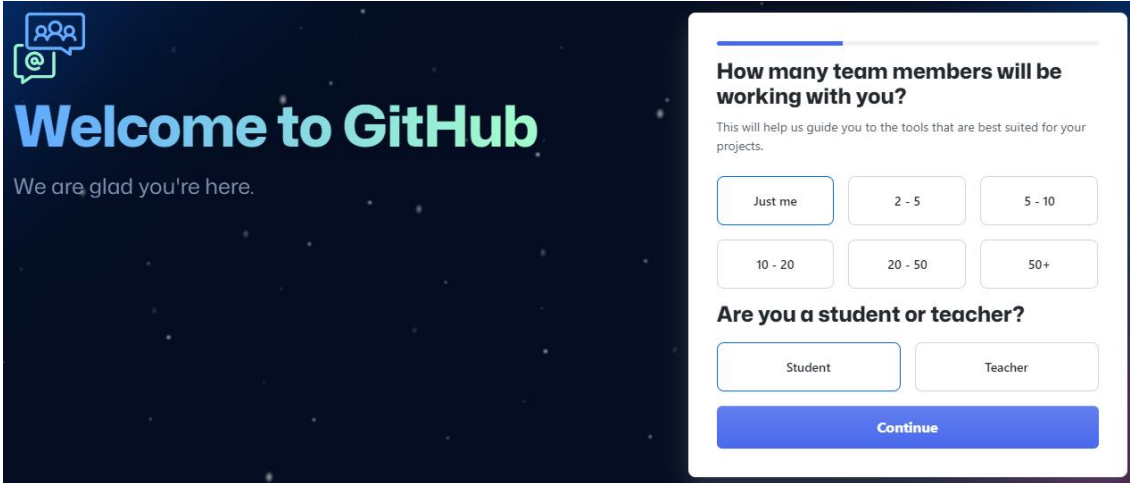
→ Enter code

Eight empty input boxes for the launch code.

Didn't get your email? [Resend the code](#) or [update your email address](#).

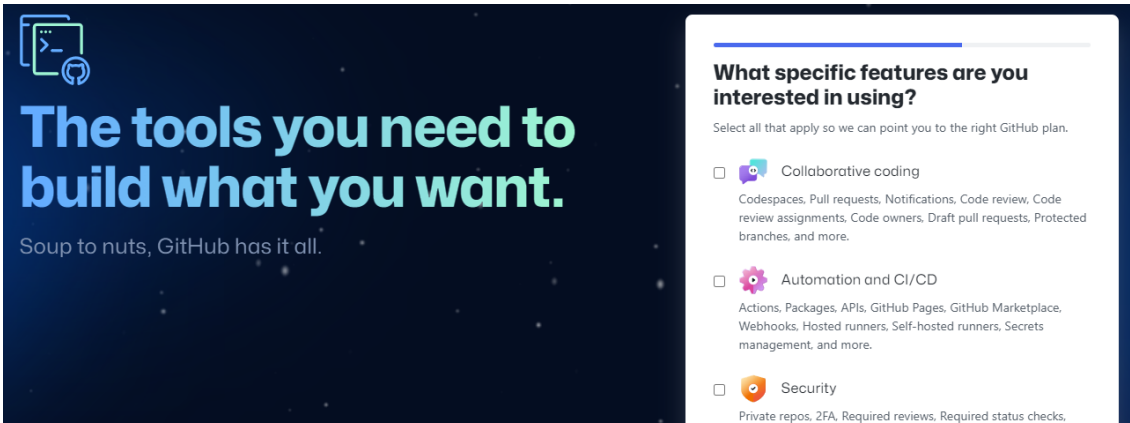
Assim sua conta já está criada e o direciona para uma página de personalização da mesma, as primeiras perguntas são quantas pessoas trabalharão com você e se você é

estudante ou professor, um exemplo de como preencher para um estudante segue na imagem.



The image shows the GitHub welcome screen. On the left, there's a dark blue background with a light blue icon of three people and a speech bubble. The text "Welcome to GitHub" is in large, bold, light blue letters, and "We are glad you're here." is in smaller, white letters below it. On the right, there's a white panel with a blue border. It contains a section titled "How many team members will be working with you?" with a subtext "This will help us guide you to the tools that are best suited for your projects." Below this are six buttons: "Just me", "2 - 5", "5 - 10", "10 - 20", "20 - 50", and "50+". Another section titled "Are you a student or teacher?" has two buttons: "Student" and "Teacher". At the bottom of the panel is a large blue button labeled "Continue".

Continuando é questionado para que o GitHub será utilizado, é só selecionar os itens que se identifica ou, até mesmo, não selecionar nada e continuar (o botão de continuar está na parte inferior da tela).

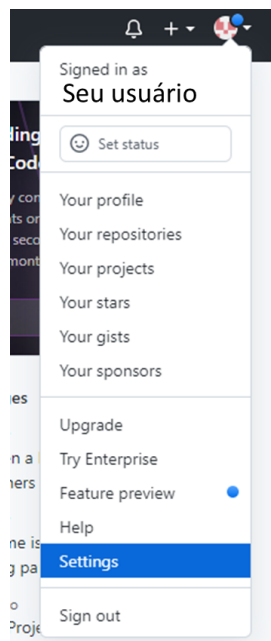


The image shows the GitHub feature selection screen. On the left, there's a dark blue background with a light blue icon of a code editor and a GitHub logo. The text "The tools you need to build what you want." is in large, bold, light blue letters, and "Soup to nuts, GitHub has it all." is in smaller, white letters below it. On the right, there's a white panel with a blue border. It contains a section titled "What specific features are you interested in using?" with a subtext "Select all that apply so we can point you to the right GitHub plan." Below this are three checkboxes with corresponding icons and text: "Collaborative coding" (with a speech bubble icon), "Automation and CI/CD" (with a gear icon), and "Security" (with a shield icon). Each checkbox has a list of features below it.

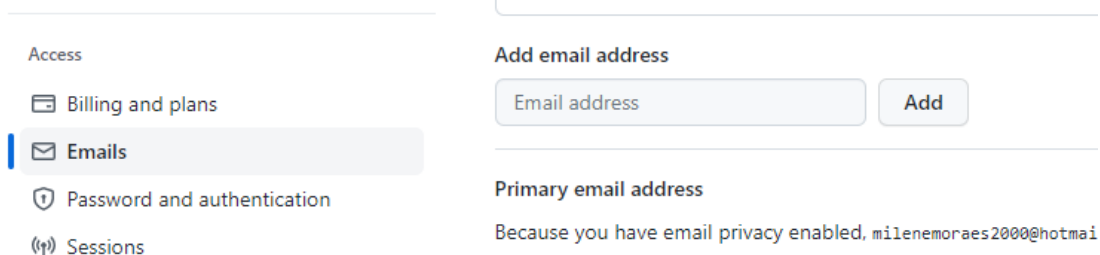
Feito isso é questionado se quer utilizar a versão Pro ou a versão gratuita, fica a decisão para o usuário (é possível conseguir os benefícios de estudante por um tempo limitado caso tenha conta institucional – isso é feito com a adição de um e-mail secundário após a seleção da opção gratuita).

1.2- Adicionar e-mail de estudante:

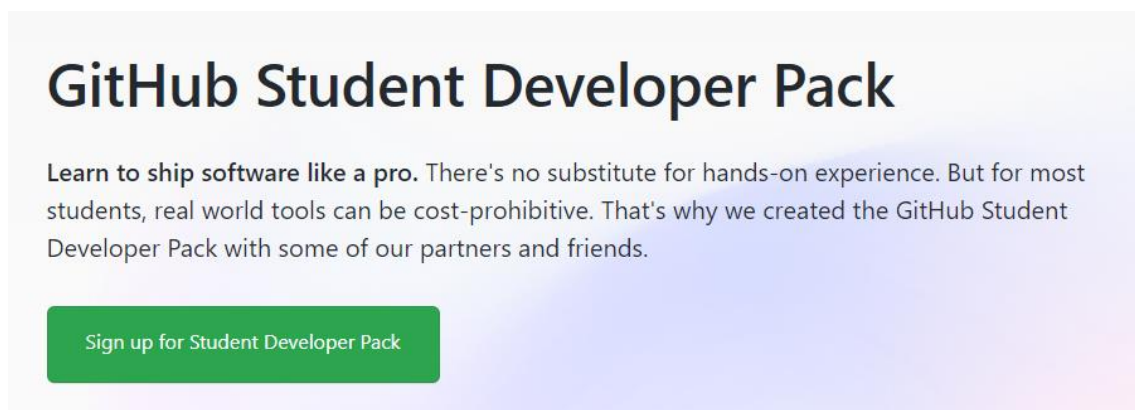
Para adicionar o e-mail de estudante você deve: Selecionar a imagem no canto superior direito e selecionar Settings (configurações).



Feito isso selecione o item Emails e Adicione seu e-mail de estudante no item Add email address.



Ao adicionar um novo e-mail será encaminhado um e-mail de verificação para o mesmo. Com o E-mail verificado é só acessar <https://education.github.com/pack> e selecionar o botão verde, como na imagem na sequência, para inscrever-se no pacote.



Então deve-se selecionar a opção para alunos (Students).

Individuals

Students

Learn using real-world development tools

- ✓ **FREE** GitHub Pro while you are a student
- ✓ Valuable GitHub Student Developer Pack partner offers
- ✓ GitHub Campus Expert training for qualified applicants

Get student benefits

Teachers

Teach your students with the industry-standard tools

- ✓ **FREE** GitHub Team for courses, coding clubs, and nonprofit research
- ✓ GitHub Classroom for managing assignments

Get teacher benefits

Feito isso é só verificar as informações e adicionar algo no campo que pergunta como você planeja utilizar o GitHub (como exemplo abaixo).


How do you plan to use GitHub? *

Estudar

Ainda é necessário adicionar uma prova que é estudante da instituição informada (como um comprovante de matrícula) no campo mostrado na imagem na sequência.

Please upload proof of your academic status.

Snap a picture of your qualifying proof of current academic status using your HD webcam or smartphone camera.



Use your computer's camera to take a picture now.

Take a picture

Already have an image? We accept JPEGs of at least 1024x768 resolution between 100KB and 10MB in size.

Upload an image

Bem como selecionar o botão verde no final da página para proceder com a aplicação.

Process my application

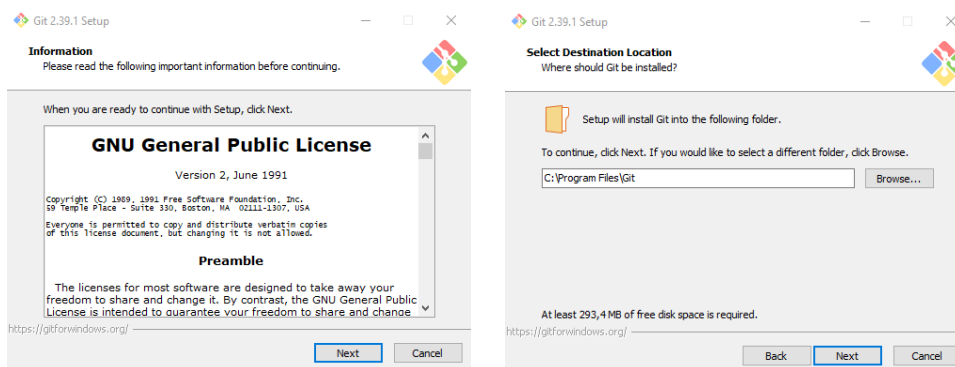
Cancel

2- Instalar o GIT (GitBash):

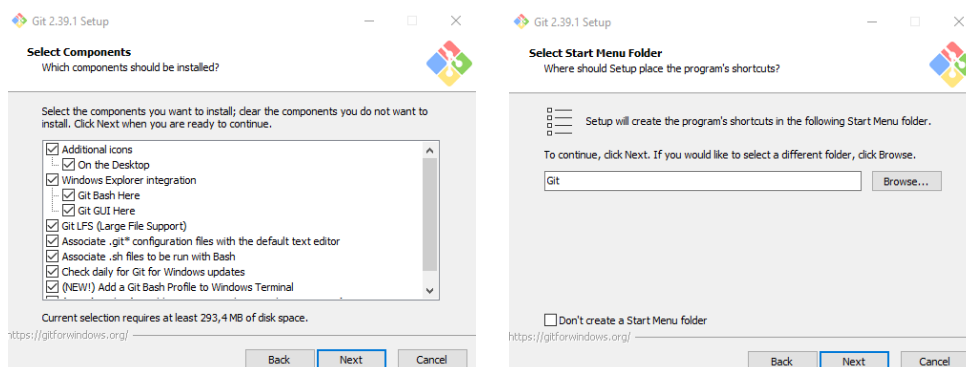
Primeiramente acessar <https://git-scm.com/download/win> e selecione a opção compatível com seu computador, no caso do exemplo abaixo 64-bit (com a seta em vermelho).

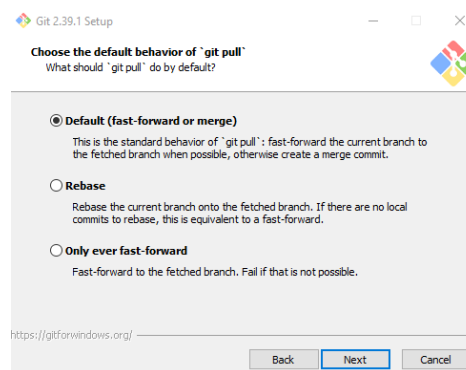
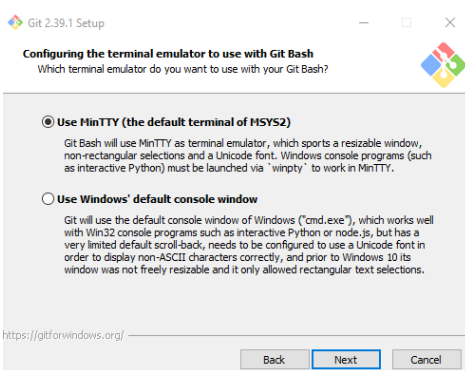
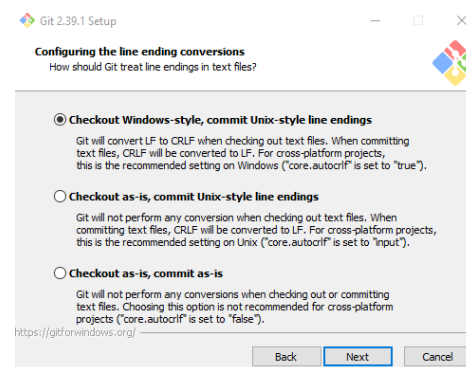
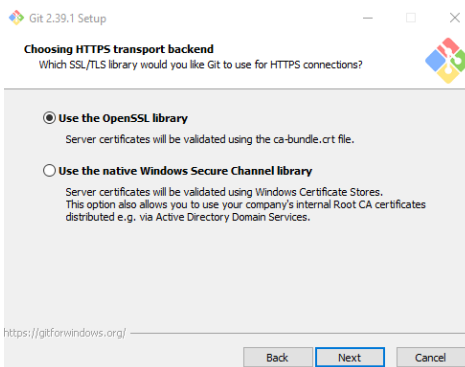
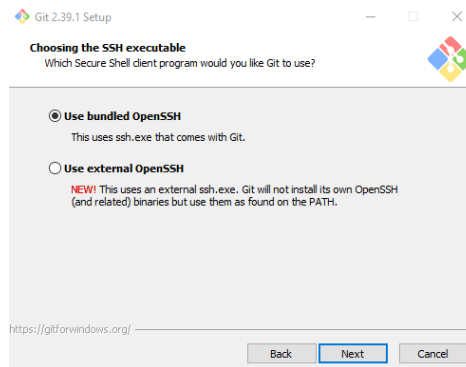
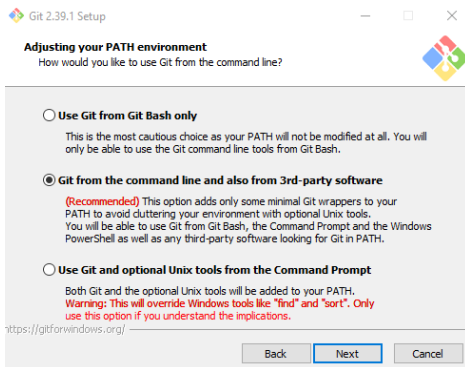
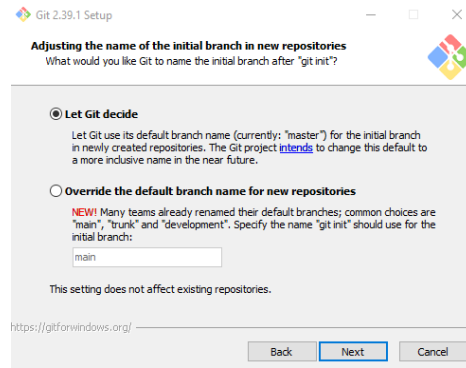
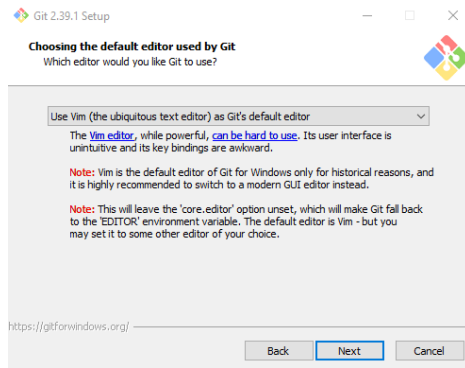


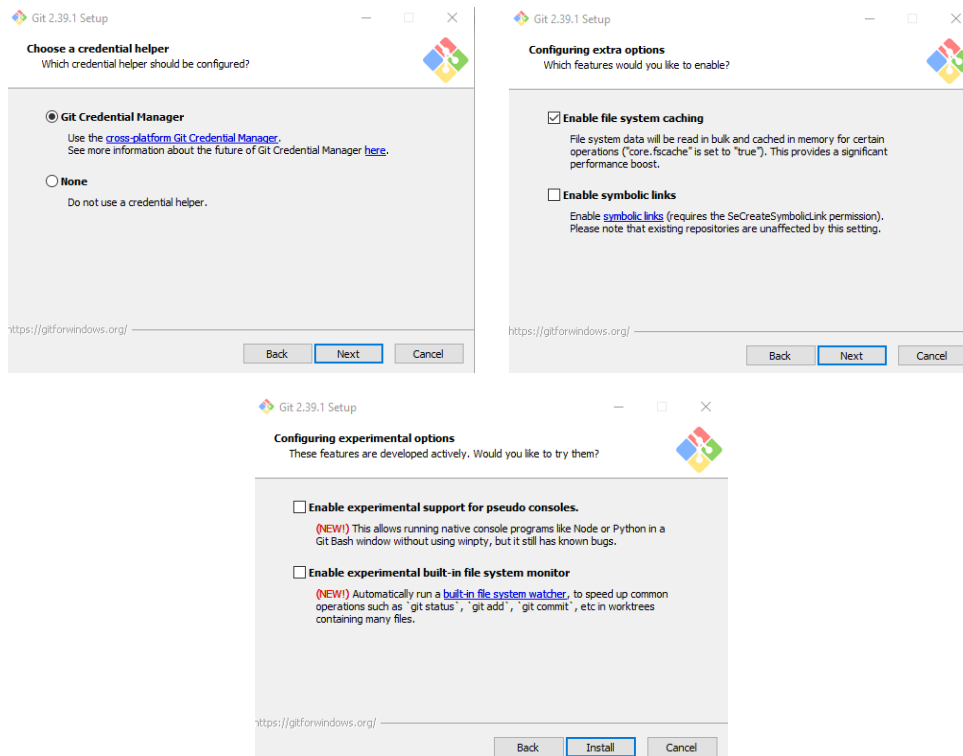
Feito isso é necessário abrir o programa baixado e ir selecionar Next, conforme as imagens abaixo.



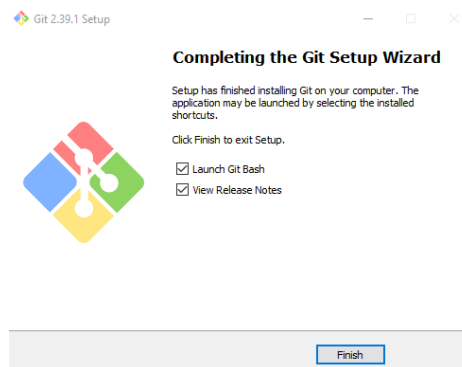
Além disso é interessante selecionar todos os itens na parte de componentes, continuando o download com o Next, conforme imagens abaixo e, por último, Install.







É possível selecionar Launch Git Bash, para abrir o Git Bash diretamente com a instalação assim que selecionar Finish, como na imagem abaixo.



Com isso será aberta uma aba onde digitamos “git --version” para identificar a versão instalada. No caso do exemplo abaixo foi a versão 2.39.1.



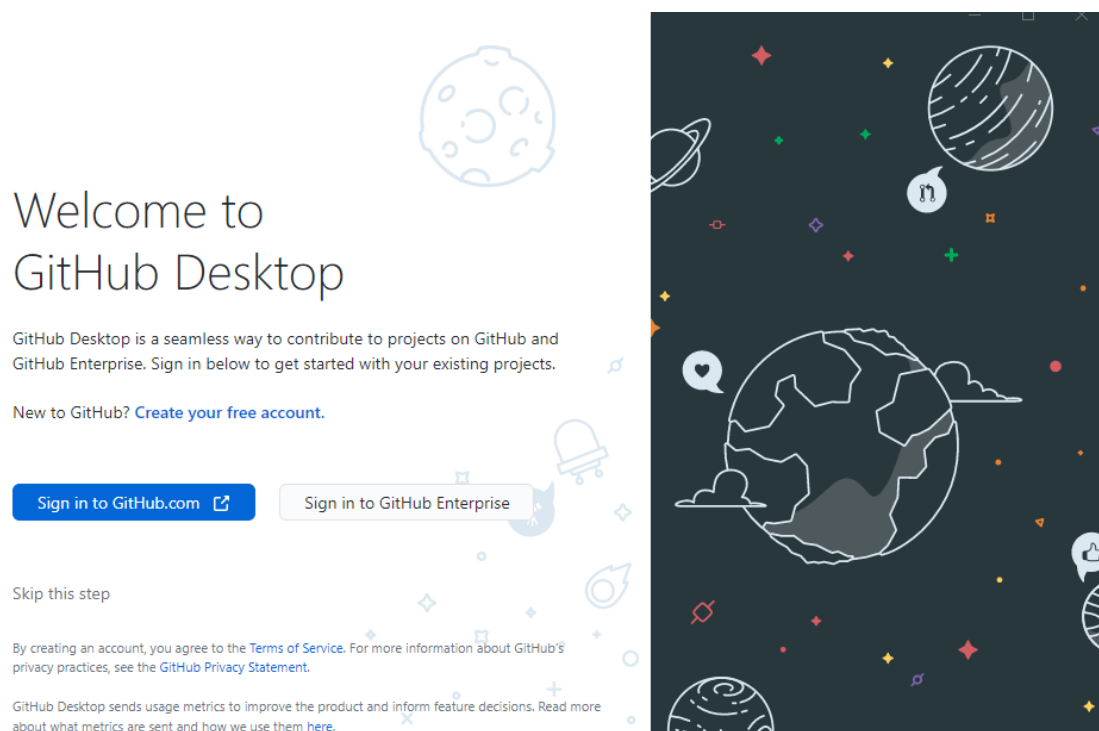
Com isso foi instalado o GitBash.

3- Instalar o GitHub Desktop:

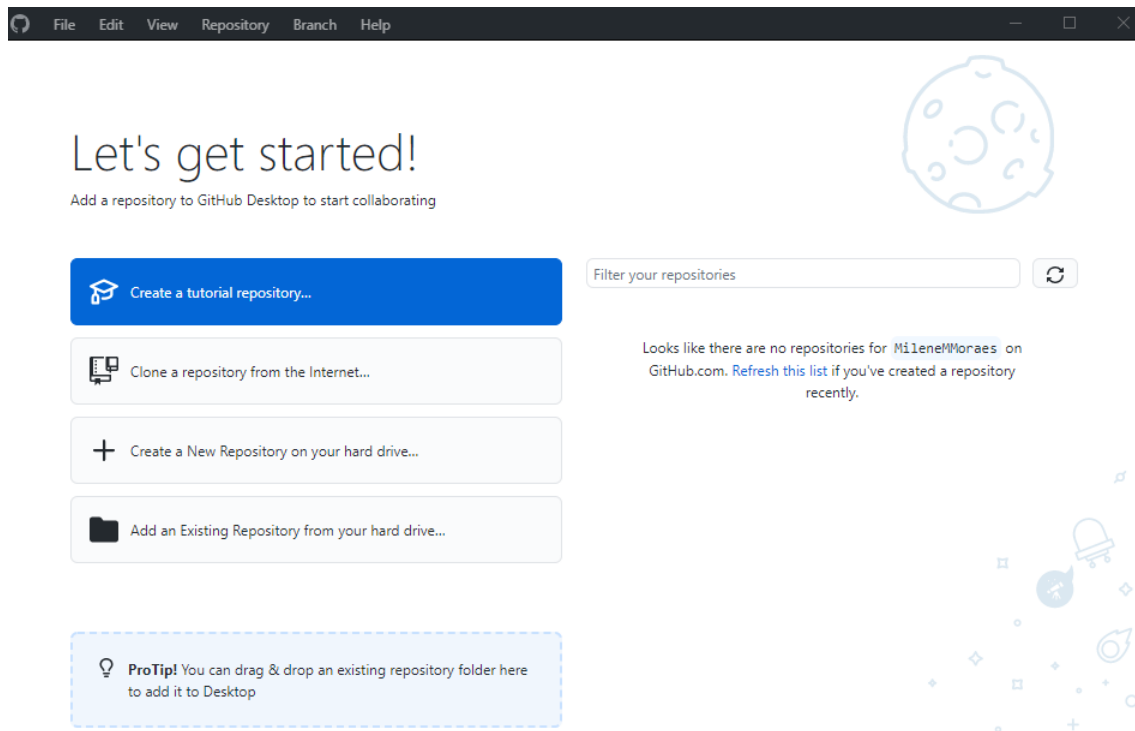
Para instalar o GitHub Desktop deve-se acessar <https://desktop.github.com/> e selecionar para baixar para Windows (botão roxo na imagem na sequência).



Feito isso é necessário abrir o arquivo baixado e conectar o mesmo com sua conta GitHub, para isso é necessário selecionar “Sign in to GitHub.com” (botão azul da próxima imagem).



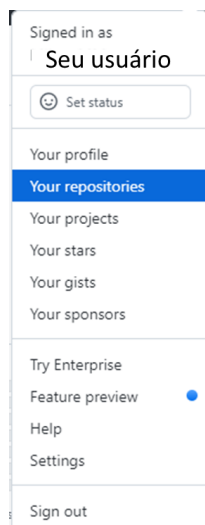
Feito isso uma aba abrirá no navegador, onde é só logar e, depois, voltar ao GitHub Desktop, onde deverá ser selecionado seu e-mail principal e “Finish”, com isso está finalizado, aparecendo na tela algo como a próxima imagem:



Assim GitBash e GitHub Desktop para Windows foram instalados.

4- Configurar o repositório:

No GitHub online ir em seu repositório no menu do canto superior direito.





Deve-se selecionar novo (botão verde “New”) para criar um novo repositório.



É necessário dar um nome para o repositório.

Repository name *

Caso queira que outras pessoas possam acessar o repositório é necessário manter o mesmo como public.

- ☒  **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐  **Private**
You choose who can see and commit to this repository.

É recomendável adicionar o README no repositório (geralmente utilizado para mostrar como baixar o repositório).

Initialize this repository with:

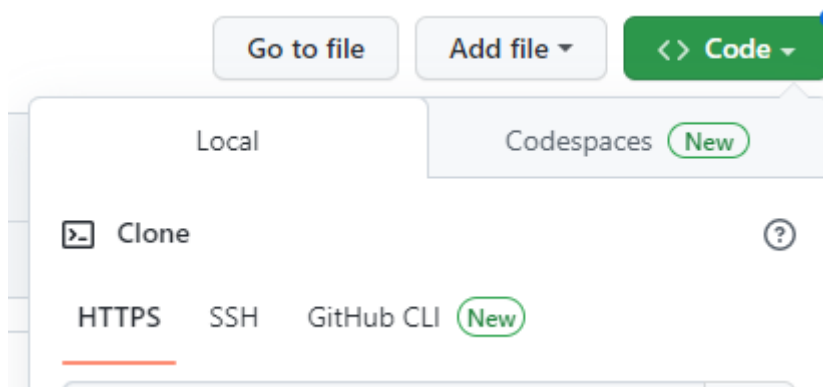
Skip this step if you're importing an existing repository.

- ☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

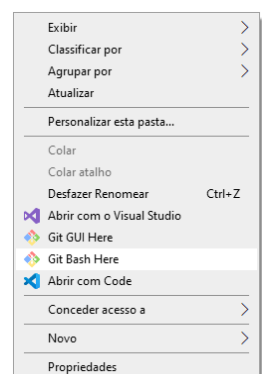
Selecione “Create repository” para criar o repositório.

Create repository

Agora queremos baixar esse repositório para máquina, para isso podemos utilizar o HTTPS em Code (copiar o código para utilizar).



Para isso abrimos a pasta Windows onde o repositório ficará, clicaremos com o botão direito e em abrimos com Git Bash, como indicado na imagem, o que abrirá o Git Bash diretamente na pasta desejada.



Então escrevemos “git clone *linkDoRepositório*” no GitDash (obs.: Para colar é necessário utilizar o botão direito do mouse (e selecionar Paste) ou utilizar Shift+Ins, dando Enter a pasta já deve ter clonado a do GitHub).

```
$ git clone https://github.com/ Continuação do link .git
```

A partir disso pode criar pastas e colocar arquivos dentro dessa pasta, como exemplo abaixo (obs.: Já é necessário ter arquivos dentro das pastas para que essas possam ser encaminhadas para o repositório – e aparecer no GitHub).

📁 Módulo 01	12/02/2023 15:29	Pasta de arquivos
📁 Módulo 02	12/02/2023 15:29	Pasta de arquivos
📁 Módulo 03	12/02/2023 15:29	Pasta de arquivos
📄 README	12/02/2023 15:24	Arquivo Fonte Ma...

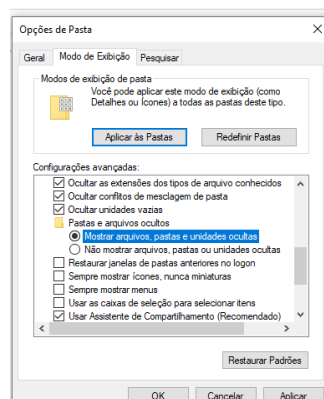
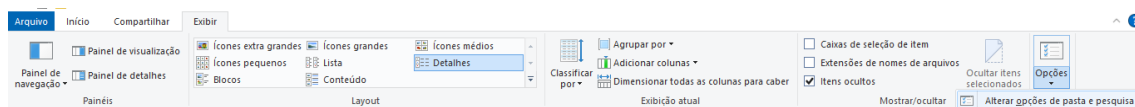
Feito isso digitamos “ls” no Git Bash para listar as subpastas da pasta que estou presente, para acessar esta é só dar o comando “cd nomedapasta/”, como o exemplo na sequência.

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec
$ ls
floripamaistec/

USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec
$ cd floripamaistec/

USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$
```

Para permitir a visualização de pastas e arquivos ocultos deve-se voltar na pasta, ir em Exibir -> Opções -> Alterar opções de pasta e pesquisa e selecionar a opção “Mostrar arquivos, pastas e unidades ocultas” (e Aplicar), como nas imagens abaixo.



Ainda é necessário confirmar o nome de usuário e o e-mail (com atenção para colocar o mesmo do GitHub) utilizando ‘git config --global user.name "preencher o seu user name” e ‘git config --global user.email “preencher com o seu e-mail”’.

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git config --global user.name "User name aqui"

USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git config --global user.email "E-mail aqui"
```

É possível verificar se as modificações foram realizadas com um 'git config --list' (para sair do mesmo é necessário apertar "q" no teclado).

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git config --list
```

Feito isso pode-se escrever 'git init' para inicializar o repositório, 'git pull' para realizar as mudanças online na máquina e 'git status' para verificar se o repositório identificou os arquivos diferentes dentro da pasta.

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git init
```

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git pull
Already up to date.
```

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git status
```

Feito isso é necessário realizar os seguintes comandos para atualizar o repositório online de acordo com a máquina: 'git add .' para atualizar todos os arquivos para área de preparação; 'git commit -m "Escreva seu comentário aqui"' para comentar a publicação com o que foi atualizado; e 'git push' para atualizar no repositório (online).

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git add .
```

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git commit -m "Meu primeiro commit"
```

```
USUARIO@DESKTOP-BUCNFBQ MINGW64 ~/Desktop/floripamaistec/floripamaistec (main)
$ git push
```

Com esses passos o repositório está configurado e atualizado.