

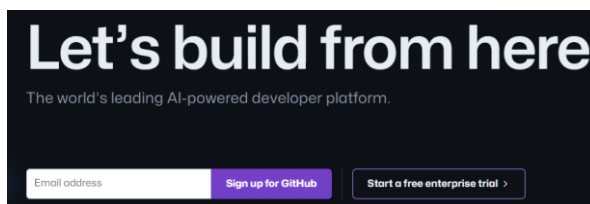
## INTRODUÇÃO AO GitHub

### Conceitos do GitHub

O GitHub é uma plataforma de hospedagem de código-fonte baseada na web que utiliza o sistema de controle de versão Git. Ele fornece uma interface amigável para colaboração em projetos de desenvolvimento de software, permitindo que os desenvolvedores compartilhem, colaborem e contribuam para projetos de código aberto ou privados.

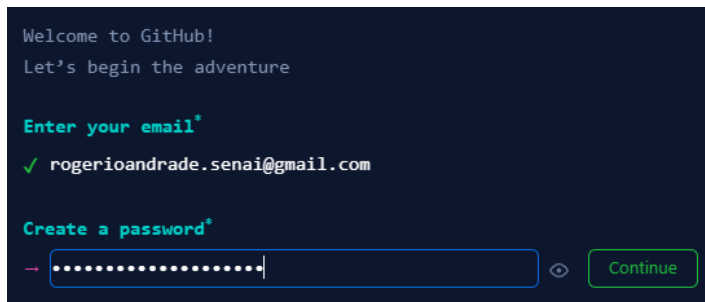
1. **Repositório (Repository):**
  - Um repositório Git é onde todos os dados e histórico de um projeto são armazenados.
  - Existem dois tipos de repositórios: repositório local (no seu computador) e repositório remoto (em um servidor).
2. **Branch:**
  - Um branch no Git é uma ramificação do seu projeto. Ele permite que você trabalhe em recursos ou correções de bugs sem afetar o branch principal.
  - O branch principal geralmente é chamado de "master" ou "main".
3. **Commit:**
  - Um commit é uma "snapshot" de um estado específico do seu projeto. Ele registra as alterações feitas nos arquivos do projeto em um determinado momento.
  - Cada commit tem uma mensagem associada que descreve as alterações feitas.
4. **Merge:**
  - O merge é o processo de combinar as alterações de um branch para outro. Geralmente é usado para mesclar um branch de desenvolvimento de volta para o branch principal.
5. **Clone:**
  - Clonar um repositório Git cria uma cópia local completa do repositório em seu computador.
  - Isso permite que você trabalhe no projeto localmente e contribua para ele.
6. **Pull:**
  - O comando pull é usado para baixar e mesclar as alterações de um repositório remoto para o seu repositório local.
  - Isso garante que você tenha as alterações mais recentes antes de trabalhar no projeto.
7. **Push:**
  - O comando push é usado para enviar seus commits locais para um repositório remoto.
  - Isso torna suas alterações disponíveis para outros colaboradores do projeto.

1 -Acessar <https://github.com>:



2 – Selecionar “Sign up for GitHub”.

3 – Entre com seu e-mail. Selecionar continue e definir senha de acesso.



Welcome to GitHub!  
Let's begin the adventure

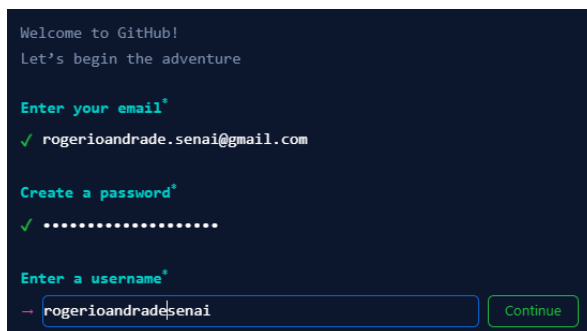
Enter your email\*

✓ rogerioandrade.senai@gmail.com

Create a password\*

– [password field] [Continue]

4- Selecionar “continue”. Defina um usuário. Selecionar “continue”.



Welcome to GitHub!  
Let's begin the adventure

Enter your email\*

✓ rogerioandrade.senai@gmail.com

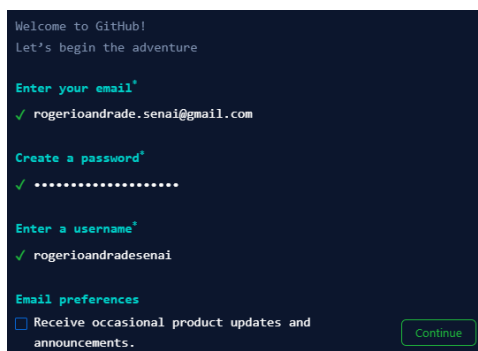
Create a password\*

✓ [password field]

Enter a username\*

– rogerioandradesenai [Continue]

5- Selecionar “continue”.



Welcome to GitHub!  
Let's begin the adventure

Enter your email\*

✓ rogerioandrade.senai@gmail.com

Create a password\*

✓ [password field]

Enter a username\*

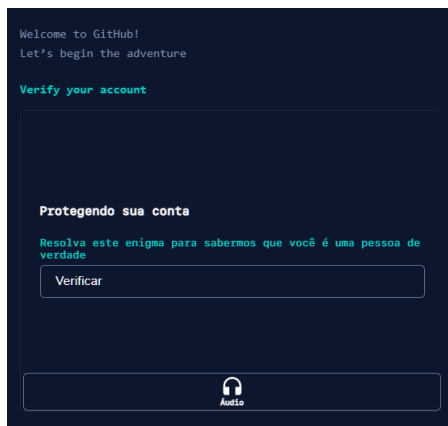
✓ rogerioandradesenai

Email preferences

☐ Receive occasional product updates and announcements.

[Continue]

## 6- Resolver o enigma:




## 7 – Resolvendo:



## 8- Informar o código enviado por e-mail.



9 – Realizar login.



Sign in to GitHub

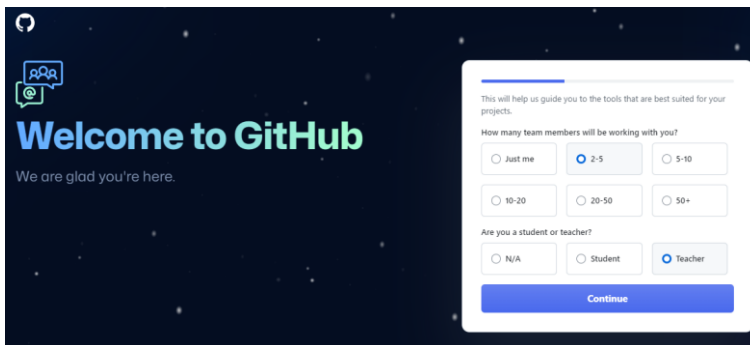
Your account was created successfully. Please sign in to continue ×

Username or email address

Password [Forgot password?](#)

Sign in

10 – Responda o questionário.



Welcome to GitHub

We are glad you're here.

This will help us guide you to the tools that are best suited for your projects.

How many team members will be working with you?

☐ Just me ☒ 2-5 ☐ 5-10

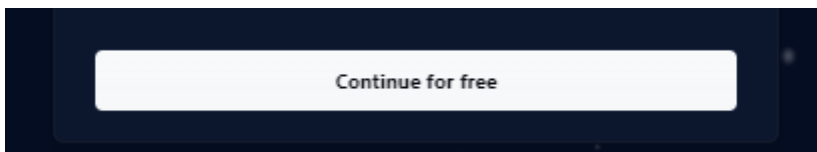
☐ 10-20 ☐ 20-50 ☐ 50+

Are you a student or teacher?

☐ N/A ☐ Student ☒ Teacher

Continue

11 – Selecionar “Continue for free”.



Continue for free

12 – Criar primeiro repositório.

### Create your first project

Ready to start building? Create a repository for a new idea or bring over an existing repository to keep contributing to it.

Create repository

[Import repository](#)


# 13 – Configurando repositório

## Create a new repository

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

Required fields are marked with an asterisk (\*).


Owner \*

 rogerioandradesenai

 / 

Repository name \*

expo


 expo is available.


Great repository names are short and memorable. Need inspiration? How about [vigilant-garbanzo](#) ?

Description (optional)

Códigos exemplos em React Expo

- ☒

 **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.

## 14 – Definir tipo de uso do código (sugestão) e selecione “Create repositor”.

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

### Choose a license

License: MIT License ▾

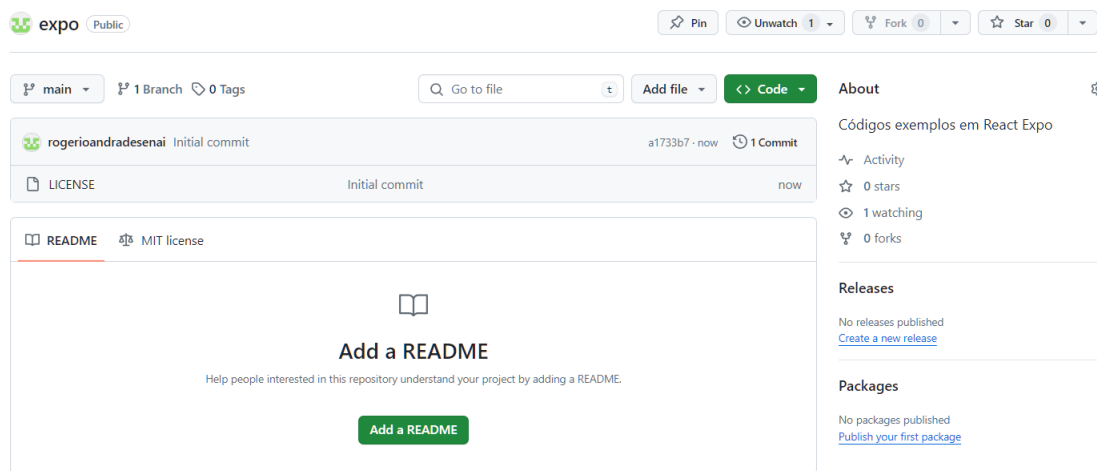
A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

 You are creating a public repository in your personal account.

Create repository

- A Licença MIT é bastante permissiva e permite que outras pessoas usem, modifiquem e redistribuam seu código, desde que incluam uma cópia do aviso de direitos autorais e a licença em qualquer redistribuição. Além disso, ela não impõe muitas restrições sobre como o software pode ser usado.

## 15- Repositório foi criado com sucesso.



## 16 0- Criar uma pasta no computador local para sincronizar com o repositório online:

```
Prompt de Comando
C:\>
Microsoft Windows [versão 10.0.19045.4474]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\admin>mkdir rogerioandrade

C:\Users\admin>cd rogerioandrade

C:\Users\admin\rogerioandrade>
```

## 17- Clonando o repositório:

```
C:\Users\admin\rogerioandrade>git clone https://github.com/rogerioandradesenai/expo
```

18 – Resultado do comando:

```
C:\Users\admin\rogerioandrade>git clone https://github.com/rogerioandradesenai/expo
Cloning into 'expo'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), done.

C:\Users\admin\rogerioandrade>
```

19 – Confirmando a clonagem:

```
C:\Users\admin\rogerioandrade>dir
O volume na unidade C é OS
O Número de Série do Volume é EC2B-0FA6

Pasta de C:\Users\admin\rogerioandrade

04/06/2024  13:23    <DIR>          .
04/06/2024  13:23    <DIR>          ..
04/06/2024  13:23    <DIR>          expo
                0 arquivo(s)                0 bytes
                3 pasta(s)      87.831.023.616 bytes disponíveis
```

- Observe que a pasta expo não existia anteriormente, é a cópia (clone) do repositório.

20 – Um arquivo novo foi adicionado à pasta expo:

```
C:\Users\admin\rogerioandrade\expo>dir
O volume na unidade C é OS
O Número de Série do Volume é EC2B-0FA6

Pasta de C:\Users\admin\rogerioandrade\expo

04/06/2024  13:58    <DIR>          .
04/06/2024  13:58    <DIR>          ..
04/06/2024  13:23                1.097 LICENSE
04/06/2024  13:57                2.384 MapView.js
                2 arquivo(s)                3.481 bytes
                2 pasta(s)      87.725.285.376 bytes disponíveis
```

21 – Adicionar o arquivo no repositório

```
C:\Users\admin\rogerioandrade\expo>git add .
warning: in the working copy of 'MapView.js', LF will be replaced by CRLF the next time Git touches it
```

## 22 – Visualizando as alterações do repositório local

```
C:\Users\admin\rogerioandrade\expo>git status
On branch main
Your branch is up to date with 'origin/main'.

Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    new file:   MapView.js
```

## 23 – Confirmando as alterações realizadas.

```
C:\Users\admin\rogerioandrade\expo>git commit -m "arquivo Mapview.js adicionado"
[main d49666f] arquivo Mapview.js adicionado
 1 file changed, 66 insertions(+)
 create mode 100644 MapView.js
```

## 24 – Sincronizando o repositório local com o remoto

```
C:\Users\admin\rogerioandrade\expo>git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 6 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 1.31 KiB | 1.31 MiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/rogerioandradesenai/expo
 18cdce9..d49666f  main -> main
```

The screenshot shows the GitHub interface for a repository named 'expo' which is public. It displays the 'main' branch with 1 branch and 0 tags. A search bar and buttons for 'Add file' and 'Code' are visible. Below, the commit history shows a single commit 'd49666f' from 1 minute ago with 4 commits in total. The commit details table lists two files: 'LICENSE' (Initial commit, yesterday) and 'MapView.js' (arquivo Mapview.js adicionado, 1 minute ago).

File	Commit Message	Commit Hash	Time
LICENSE	Initial commit	d49666f	yesterday
MapView.js	arquivo Mapview.js adicionado	d49666f	1 minute ago

## 25 – Criando um novo Branch

```
C:\Users\admin\rogerioandrade\expo>git checkout -b dev_01
Switched to a new branch 'dev_01'
```

## 26 – Visualizando o novo Branch.

```
C:\Users\admin\rogerioandrade\expo>git status
On branch dev_01
nothing to commit, working tree clean
```



- Note que após a criação do novo Branch ele é o ativo (o primeiro Branch é o principal: main). O novo Branch está vazio.

27 – Adicionando um novo arquivo ao Branch ativo.

```
Pasta de C:\Users\admin\rogerioandrade\expo
05/06/2024  08:55    <DIR>          .
05/06/2024  08:55    <DIR>          ..
04/06/2024  13:23             1.097 LICENSE
04/06/2024  13:57             2.384 MapView.js
04/06/2024  13:23             1.097 novoArquivo
               3 arquivo(s)             4.578 bytes
               2 pasta(s)      84.775.030.784 bytes disponíveis
C:\Users\admin\rogerioandrade\expo>git add novoArquivo
```

- Apenas o novoArquivo foi adicionado.

28- Verificando o status do Branch

```
C:\Users\admin\rogerioandrade\expo>git status
On branch dev_01
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   novoArquivo
```

- Atenção: para confirmar deve ser utilizado commint

```
C:\Users\admin\rogerioandrade\expo>git commit -m "atualizando novo branch"
[dev_01 735ab82] atualizando novo branch
 1 file changed, 21 insertions(+)
 create mode 100644 novoArquivo
```

29 – Alterando para o Branch principal:

```
C:\Users\admin\rogerioandrade\expo>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
```

- Agora está no branch principal

### 30 – Para ver o log de eventos realizado no Branch

```
C:\Users\admin\rogerioandrade\expo>git log
commit d49666f053d4c07efcf9fa1cf2ad0eb1dad2057e (HEAD -> main, origin/main, origin/HEAD)
Author: rogerioandradesenai <rogerioandrade.senai@example.com>
Date:   Wed Jun 5 08:41:48 2024 -0300

    arquivo Mapview.js adicionado

commit 18cdce9efd74b1de067fdce860060bc2807b0903
Author: rogerioandradesenai <rogerioandrade.senai@gmail.com>
Date:   Wed Jun 5 08:28:22 2024 -0300

    Delete MapView.js

commit 94eddf8190d96139b0706412b10faedf978398df
Author: rogerioandradesenai <rogerioandrade.senai@example.com>
Date:   Tue Jun 4 16:43:53 2024 -0300

    Adicionado arquivo novo

commit a1733b7dad7b8badc261af3402a6f7370726c71
Author: rogerioandradesenai <rogerioandrade.senai@gmail.com>
Date:   Tue Jun 4 11:48:16 2024 -0300

    Initial commit
```

### 31 – Para saber em qual Branch está:

```
C:\Users\admin\rogerioandrade\expo>git branch -v
dev_01 735ab82 atualizando novo branch
* main d49666f arquivo Mapview.js adicionado
```

### 32 – Mesclando Branch

```
C:\Users\admin\rogerioandrade\expo>git checkout dev_01
Switched to branch 'dev_01'

C:\Users\admin\rogerioandrade\expo>git merge main
Already up to date.
```

- Ir para o branch e em seguida mesclar com o qual deseja

### 33 – Atualizar repositório remoto a partir do local

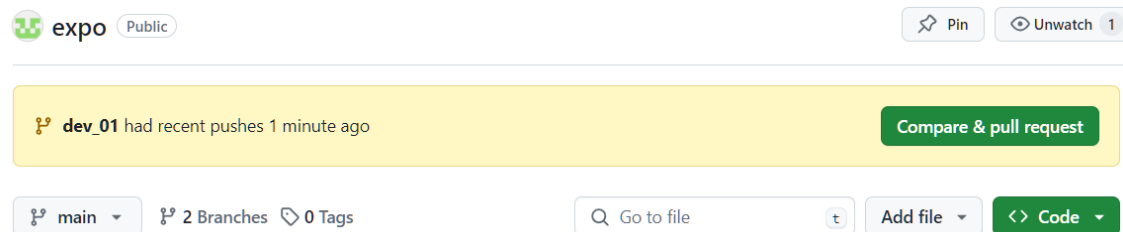
```
C:\Users\admin\rogerioandrade\expo>git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.

C:\Users\admin\rogerioandrade\expo>git push
Everything up-to-date
```

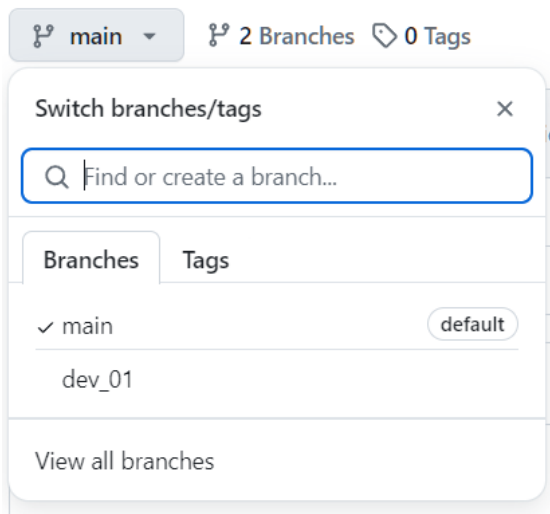
- Deve estar no Branch principal e então aplicar push

34 – Atualizar repositório remoto a partir do local garantindo que todos os Branch do local estejam no remoto

```
C:\Users\admin\rogerioandrade\expo>git push origin --all
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 6 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 295 bytes | 295.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'dev_01' on GitHub by visiting:
remote:   https://github.com/rogerioandradesenai/expo/pull/new/dev_01
remote:
To https://github.com/rogerioandradesenai/expo
 * [new branch]      dev_01 -> dev_01
```



- Observe a atualização acima



35 – Remover um Branch.

```
git branch -d meu-branch
```



