



INSTITUTO SUPERIOR POLITÉCNICO DE TECNOLOGIAS E CIÊNCIAS

ENGENHARIA DE SOFTWARE

2023/2º SEMESTRE

PRÁTICA LABORATORIAL #5

GIT E GITHUB

Elaborado por:

Cátia Adão-20201007

Avelar Manuel-

Énio Carlos-

Luanda, 12/05/2023

INTRODUÇÃO

GitHub é uma plataforma de hospedagem de código-fonte e arquivos com controle de versão usando o Git. Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou Open Source de qualquer lugar do mundo.

OBJECTIVOS DA PRÁTICA LABORATORIAL

1. Desenvolver habilidades de gestão de projetos.
2. Trabalhar em equipe e colaborar.
3. Desenvolver habilidades tecnológicas.
4. Promover a aprendizagem ativa e o engajamento.
5. Desenvolver habilidades de pensamento crítico.
6. Promover a aprendizagem autônoma.

ATIVIDADES

Passo 1

Para esta prática são necessários os seguintes requisitos:

1. Ter uma conta no Github (<https://github.com>)
2. Faça download da ferramenta Git (<https://git-scm.com/downloads>)
3. Antes de começar a usar o Git, é necessário a cessar a linha de comando do Git e configurá-lo com o seu nome de usuário e endereço de e-mail. Use os seguintes comandos no terminal para fazer isso: `git config --global user.name "Seu nome"`
`git config --global user.email seu-email@exemplo.com`

Passo 2

Criar um repositório local, utilize o comando **\$ git init**. Lembre-se de verificar se está dentro da pasta pretendida, caso não esteja utilize o comando `cd <nome da pasta ou caminho>` e a seguir pode inserir o comando acima citado.

```
MINGW64 C:/Users/Catia Adao/Documents/PraticaGit
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit
$ git init
Initialized empty Git repository in C:/Users/Catia Adao/Documents/PraticaGit/.git/
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (master)
$ |
```

Passo 3

Depois do repositório estar criado podemos adicionar arquivos, para isso utilize o comando **\$ add**. (se quiser adicionar todos os arquivos que estão na pasta do repositório), caso deseje apenas adicionar arquivos específicos utilize o comando **\$ add <nome_do_arquivo>**.

```
C/
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (master)
$ git add .
```

Para ver todos os ficheiros adicionados ao repositório utilize o comando **\$ git status**

```
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (master)
$ git add .
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (master)
$ git status
On branch master
No commits yet
Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
    new file:   CSS/estilos.css
    new file:   Js/script.js
    new file:   index.html
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (master)
$
```

Passo 4

Depois de adicionar os ficheiros no repositório é necessário confirmar, ou seja, efetivar as alterações para isso utilize o comando **\$ git commit -m** “adicione um comentário que faça sentido as alterações efetuadas”

```
Catia Adao@DESKTOP-12M1Z80 MINGW64 ~/Documents/PraticaGit (master)
$ git commit -m "Adicionei a primeira versão do Projeto. A pagina de login e o formulário de cadastro já esta concluida."
```

A seguir será impresso a confirmação dos ficheiros commitados.

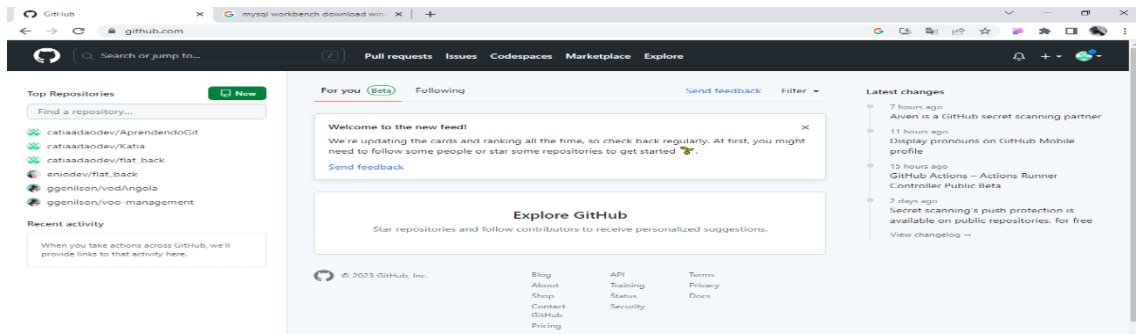
```
Catia Adao@DESKTOP-12M1Z80 MINGW64 ~/Documents/PraticaGit (master)
$ git commit -m "Adicionei a primeira versão do Projeto. A pagina de login e o formulário de cadastro já esta concluida."
[master (root-commit) 0205c45] Adicionei a primeira versão do Projeto. A pagina de login e o formulário de cadastro já esta concluida.
3 files changed, 17 insertions(+)
 create mode 100644 CSS/estilos.css
 create mode 100644 Js/script.js
 create mode 100644 index.html

Catia Adao@DESKTOP-12M1Z80 MINGW64 ~/Documents/PraticaGit (master)
$ |
```

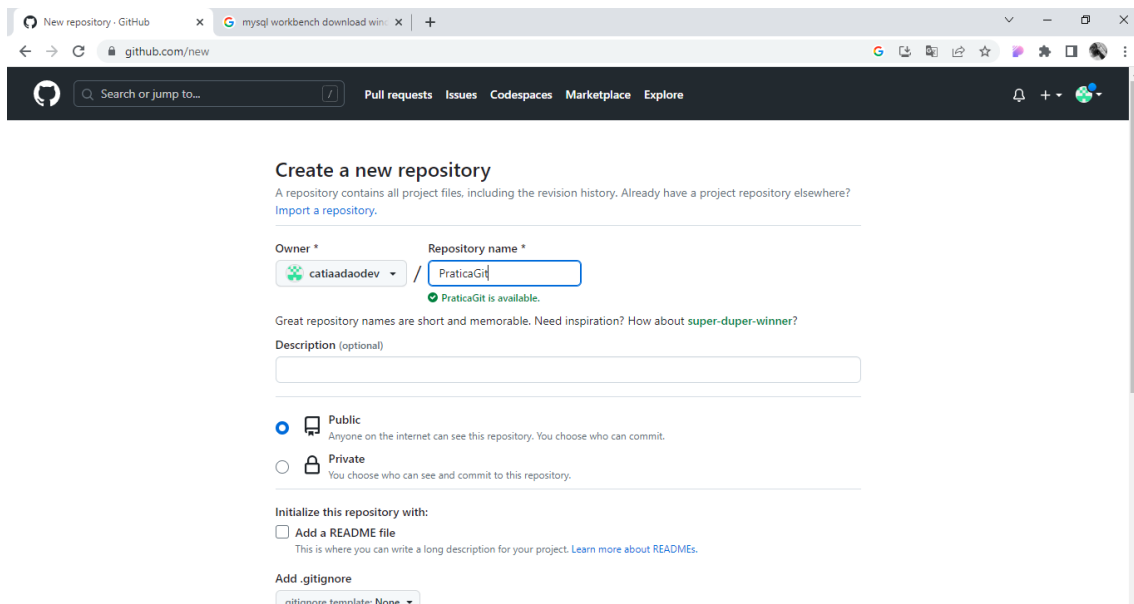
Passo 5

Depois de já ter criado o seu repositório local e ter adicionado e commitado os seus ficheiros é importante ter também estes ficheiros guardados na nuvem (repositório remoto), para isso basta a cessar a sua conta no Github e seguir o seguintes passos:

1. Clique em New



2. Deia um nome ao repositório



New repository · GitHub

mysql workbench download win

github.com/new

Search or jump to...

Pull requests Issues Codespaces Marketplace Explore

Create a new repository

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

Owner * catiaadaodev / Repository name * PraticaGit

Great repository names are short and memorable. Need inspiration? How about [super-duper-winner?](#)

Description (optional)

☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.

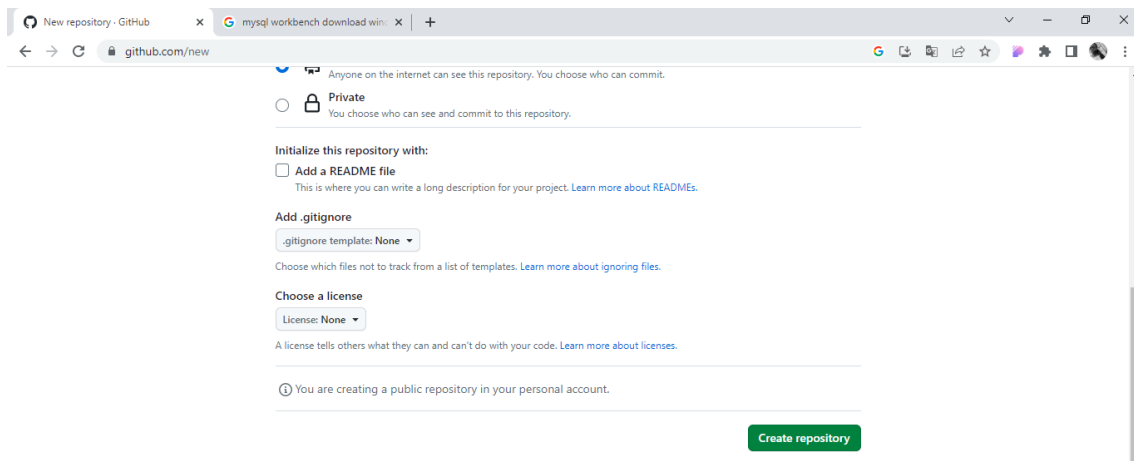
☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:

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

Add .gitignore
.gitignore template: None

3. Selecione o tipo de visualização, Público (caso pretenda que outras pessoas visualizem o seu projeto) ou Privado (caso pretenda que apenas você visualize o seu projeto) e confirma a criação do repositório



☒ **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.

Initialize this repository with:

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

Add .gitignore
.gitignore template: None

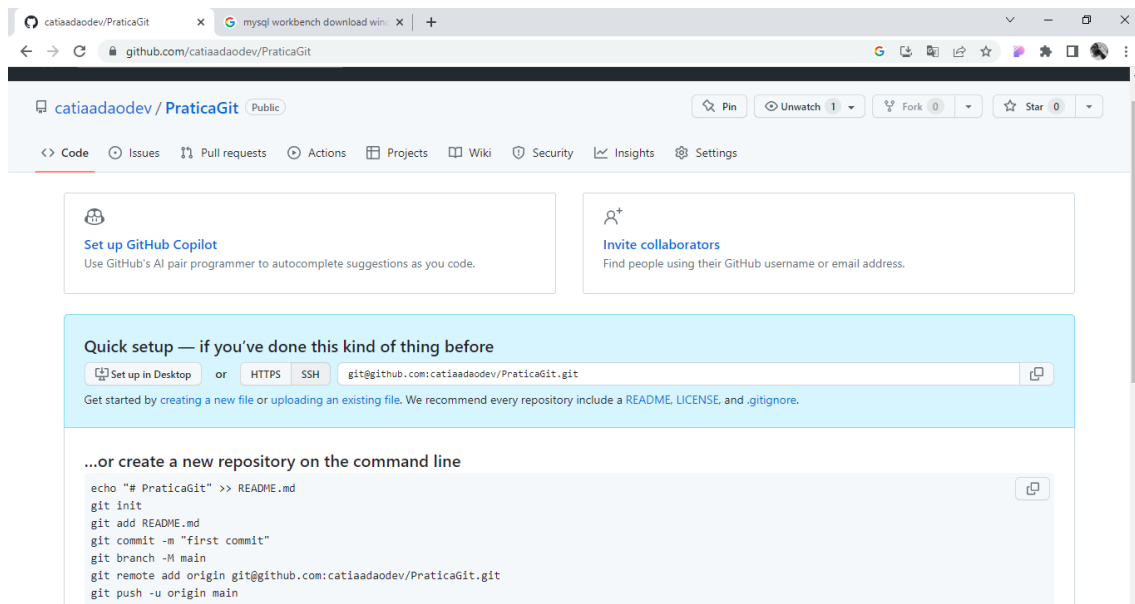
Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

Choose a license
License: None

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

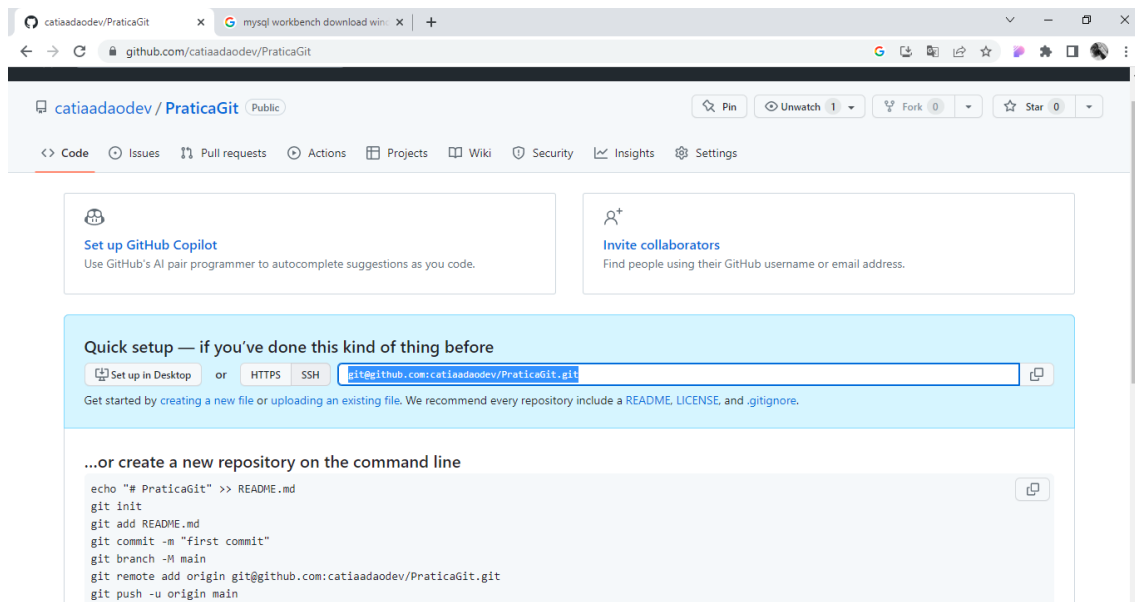
Create repository

Repositório criado com sucesso



Passo 6

Para conectar a o seu repositório remoto com o seu repositório local (criado na sua máquina), selecione e copia o endereço do repositório remoto



Passo 7

Volte para linha de comando do Git utilize o comando **\$ git remote add origin <cole aqui o endereço copiado >**

```
git remote add origin git@github.com:catiaaadoev/PraticaGit.git
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ git remote add origin git@github.com:catiaaadoev/PraticaGit.git
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
```

Passo 8

Depois de conectar os seus repositórios (local e remoto), mande para o repositórios remoto os ficheiros guardados no repositório local (de formas que mesmo que aconteça algo com a sua máquina não terá que preocupar-se pois, terá tudo guardado na nuvem), para isso utilize o comando **\$ git push -u origin main**

```
git push -u origin main
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (7/7), 744 bytes | 74.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:catiaaadoev/PraticaGit.git
 * [new branch]    main -> main
Branch 'main' set up to track 'origin/main'.
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
```

Passo 9

Sempre que alterar algum ficheiro é importante verificar os status **\$ git status**

```
git status
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ git status
```

Os ficheiros modificados aparecerão em vermelho

```
git status
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (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 committed)
  (use "git restore <file>..." to discard changes in working directory)
        modified:   CSS/estilos.css
        modified:   index.html

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        CSS/js/

no changes added to commit (use "git add" and/or "git commit -a")
```

Utilize o comando **\$ add .** para adicionar todos os ficheiros modificados

```
git add .
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ git add .
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
```

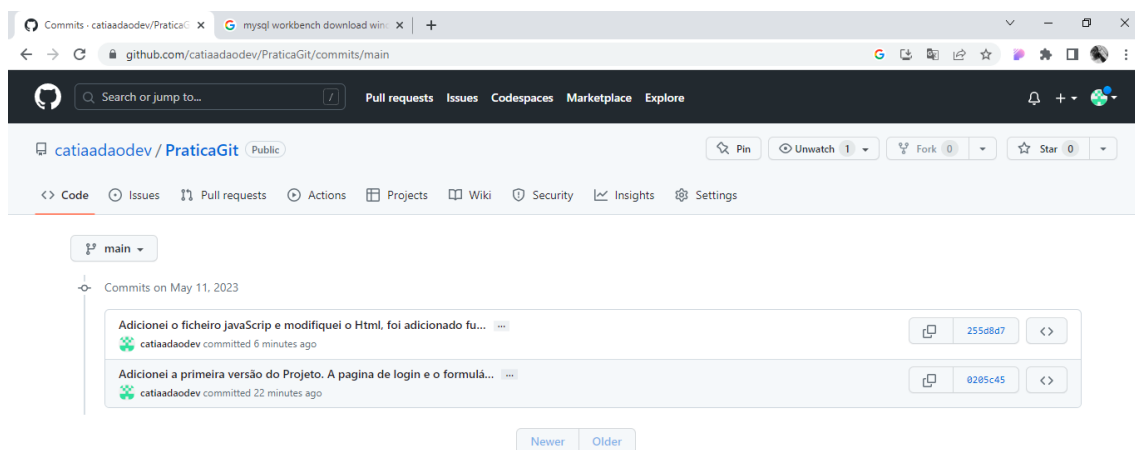
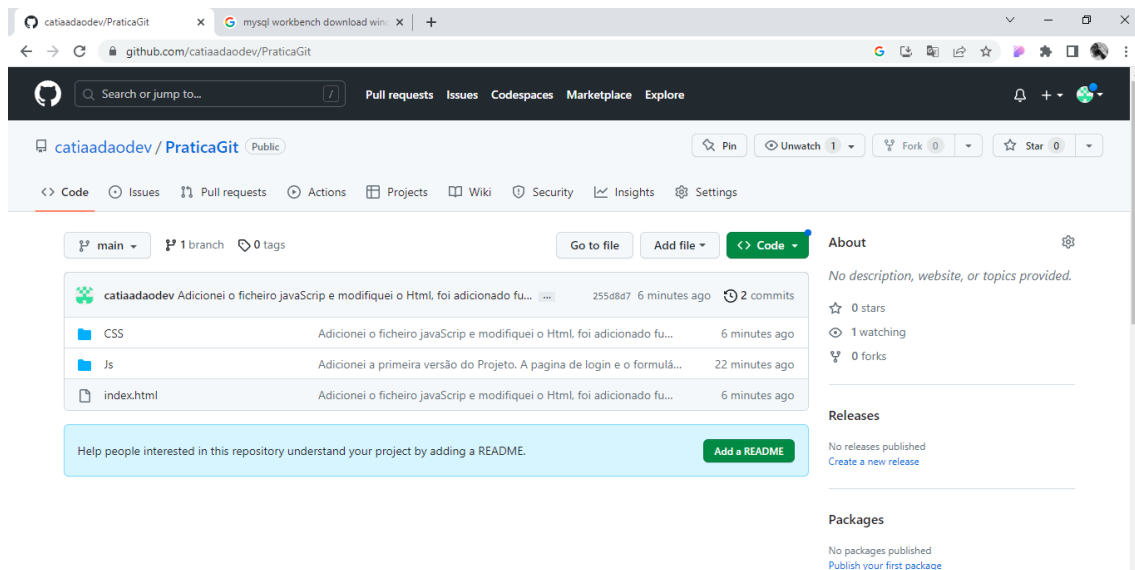
Faça a confirmação **\$ git commit -m “Mensagem que descreva as modificações feitas”**

```
git status
Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ 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:   CSS/js/script.js
        modified:   CSS/estilos.css
        modified:   index.html

Catia Adao@DESKTOP-12NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ git commit -m "Adicionei o ficheiro javascript e modifiquei o Html, foi adicionado funções como o empréstimo e devolução de livros"
[main 25d8d7] Adicionei o ficheiro javascript e modifiquei o Html, foi adicionado funções como o empréstimo e devolução de livros
3 files changed, 4 insertions(+), 1 deletion(-)
create mode 100644 CSS/js/script.js
```

Acesse ao Github para verificar os commits feitos



Passo 10

Criar uma Branch ou seja ramificação. Quando se trabalha em colaboração é intuitivo que cada elemento trabalhe em uma parte específica do projeto, a criação de branch permite que cada elemento trabalhe separadamente e sem ter que fazer alterações directamente no projeto principal. Para isso utilize o comando `$ git branch <nome_da_branch>`



Para deixar de utilizar o projeto principal e passar a trabalhar na branch criada utilize o comando `$ git checkout <nome_da_branch>`

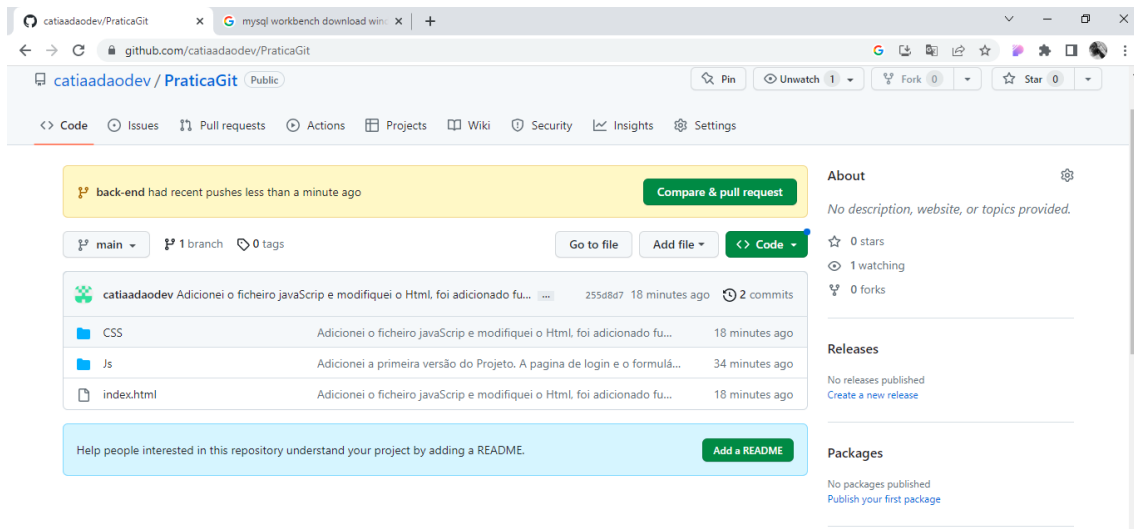

```

catia@AdaoDESKTOP-32NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ git branch back-end

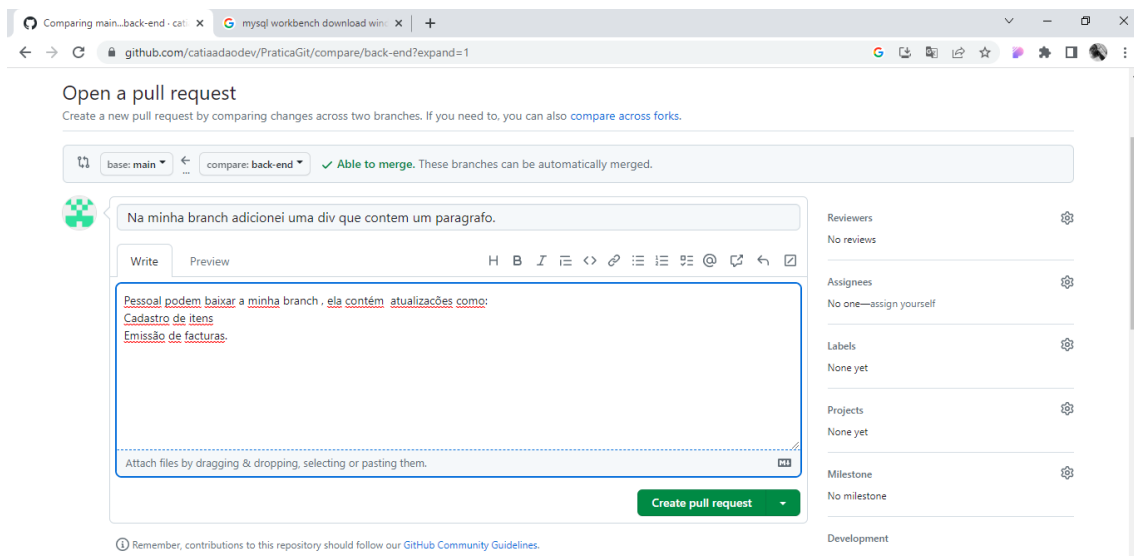
catia@AdaoDESKTOP-32NM2B0 MINGW64 ~/Documents/PraticaGit (main)
$ git checkout back-end
Switched to branch 'back-end'

```

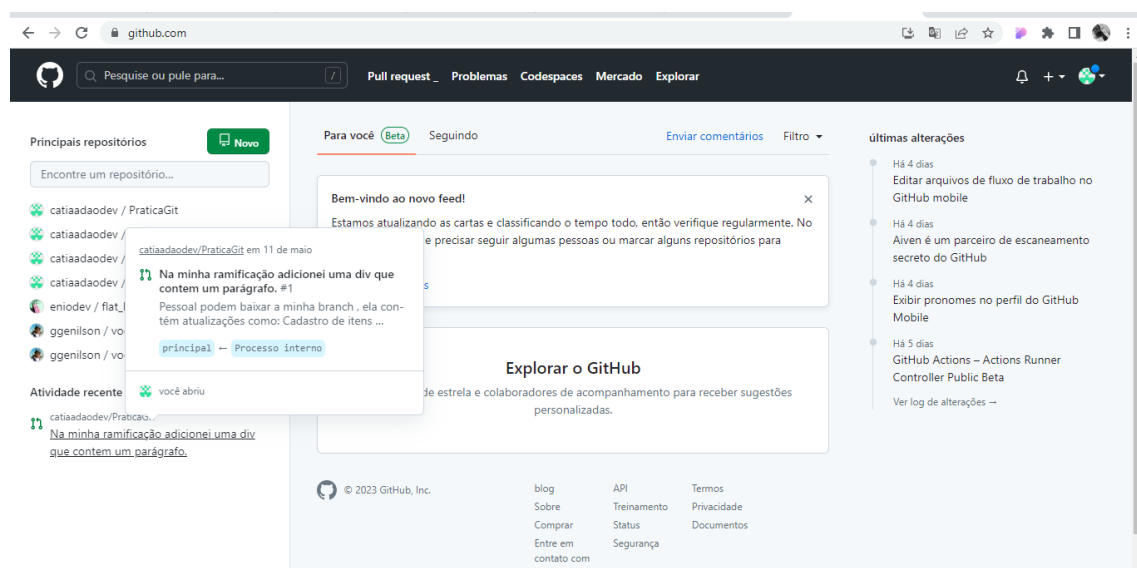
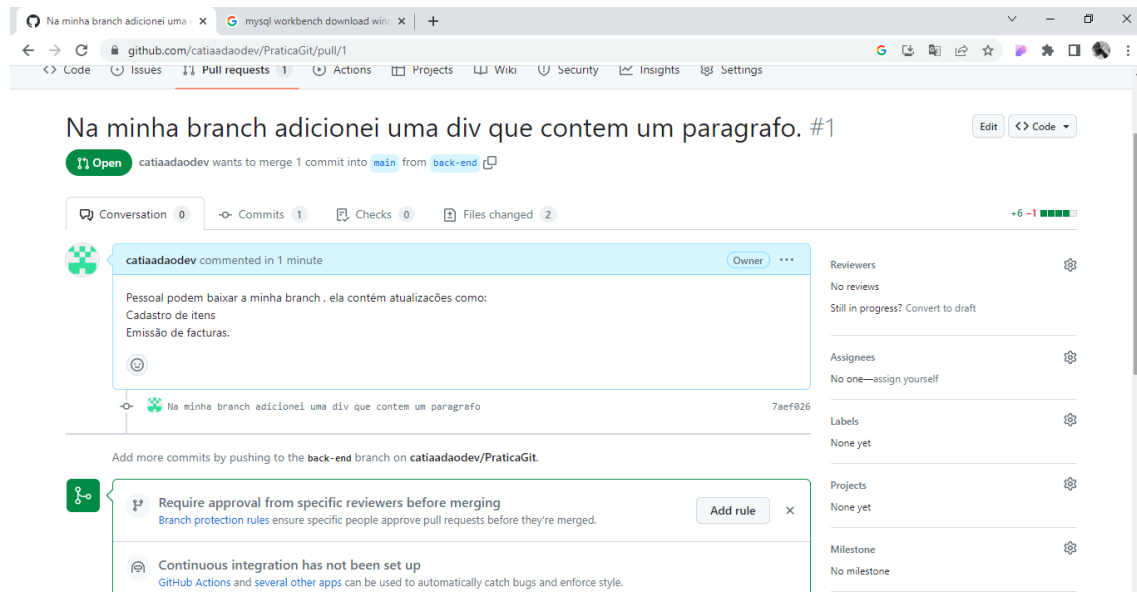
Volte para o Github e verificará a branch criada, clique em Compare e pull request



Agora é só adicionar uma mensagem para que os outros membros possam saber que partes do projeto foram adicionadas ou modificadas



Clique em Create pull request para confirmar, e está feito! Branch criada com sucesso



Link do repositório: [catiaadaodev/PraticaGit \(github.com\)](https://github.com/catiaadaodev/PraticaGit)