

TRABALHO COM O GITHUB

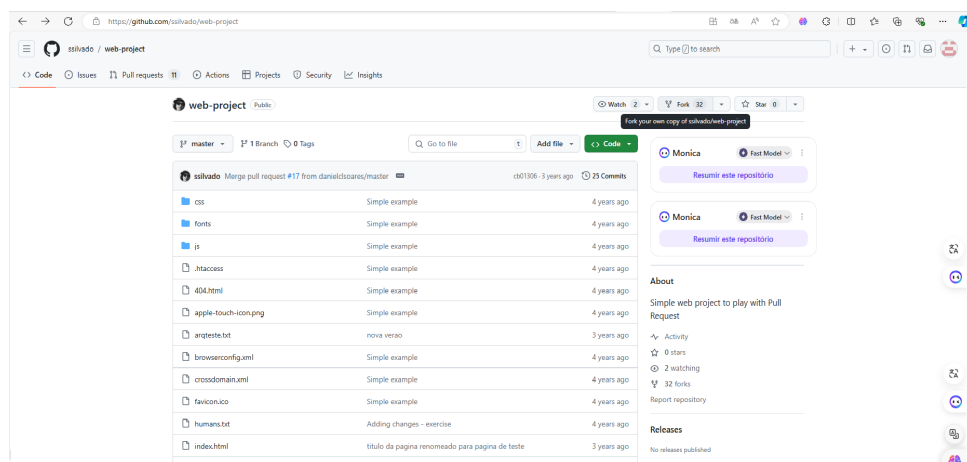
Professores: Dilson Damião, Mauricio Thiel e Eliza Costa*Name:* João Marcos Modesto Ribeiro

TEXTO

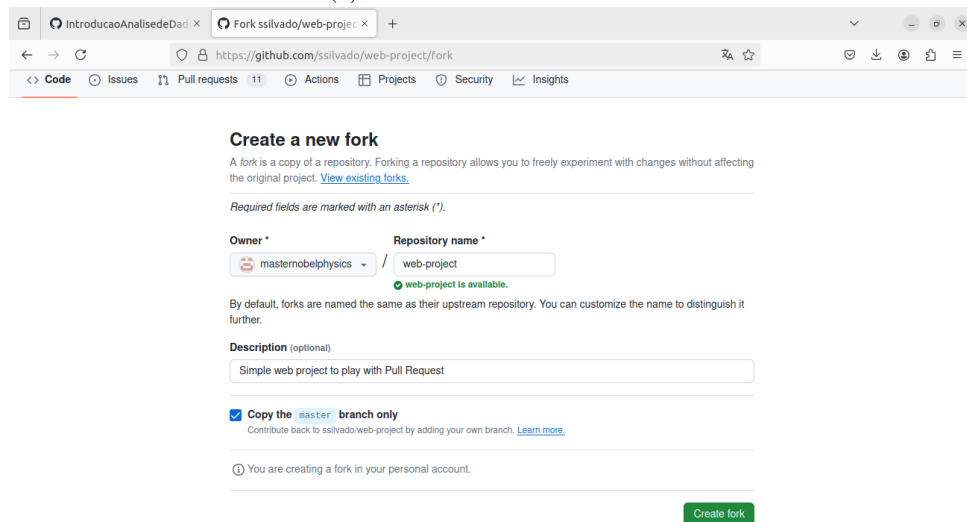
EXERCICIO GITHUB

No trabalho, vamos trabalhar com os repositórios do github desde o comando fork para o github pessoal, preparação do commit após alguma alteração até o pull request do repositório que esta sendo trabalhado .

Passo 1: Vamos no site do github no repositório: <https://github.com/ssilvado/web-project> e efetue o fork para efetuar uma copia do repositorio:



(a) Caminho do site até o fork



(b) Criando o fork

Figura 1: Caminho do site até o fork.

Após o fork, temos a seguinte janela do site, e o link toma o seguinte aspecto: <https://github.com/masternobelphysics/web-project>, sendo assim fez-se uma copia para o repositório do github pessoal como

mostrado abaixo:

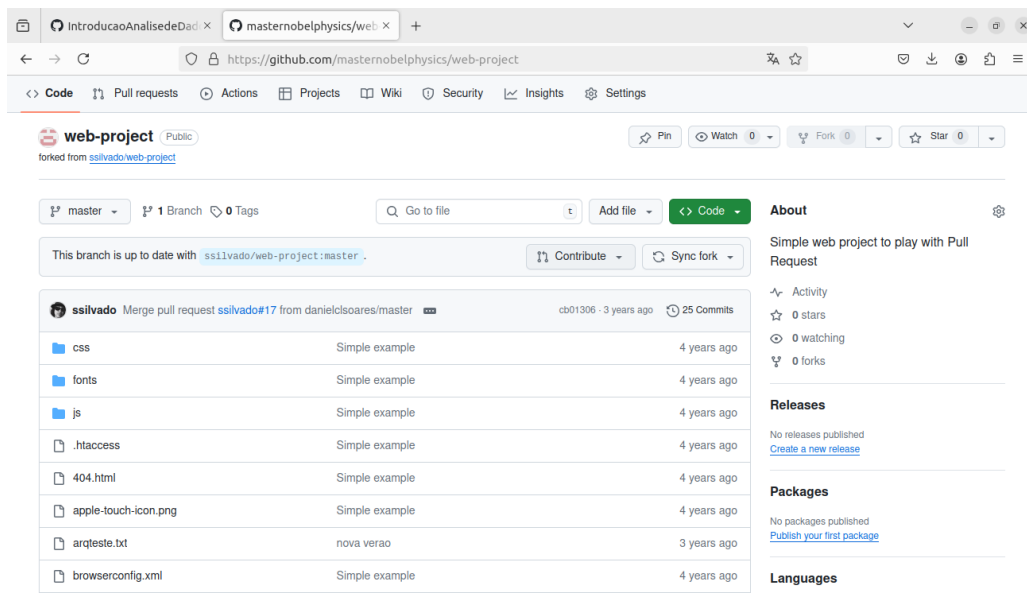


Figura 2: Aspecto do repositório github após o fork com o caminho do meu repositório pessoal

Agora vamos para o terminal linux, faremos a instalação do github com o comando `sudo apt get install git` e após ele ser instalado, usaremos os seguintes comandos em sequência:

```
1 git config --global user.name [Name]
2 git config --global user.email [Email]
3 git config --global user.github [Account]
```

Logo teremos a seguinte tela:

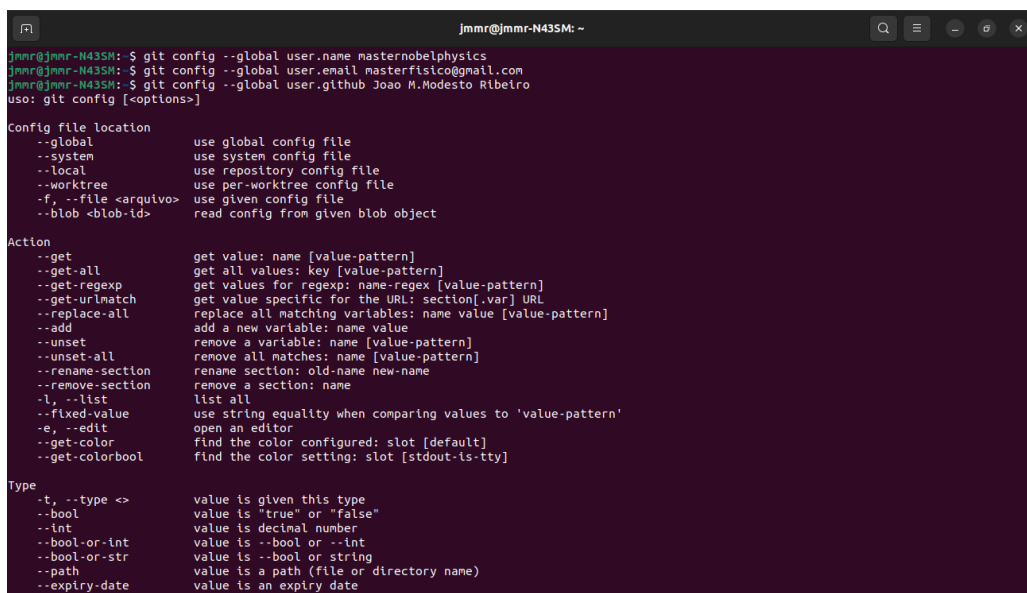


Figura 3: Execução dos comandos para a configuração.

Após isso, usaremos o comando `git clone` para efetuar a cópia do repositório do fork anterior associado para a máquina, e efetuaremos a entrada da pasta pelo terminal para monitoração de alteração em algum arquivo do repositório até a preparação e da execução do commit, logo temos:

```
1 git clone https://github.com/masternobelphysics/web-project
2 cd web-project/
3 pwd
```

```

4 ls
5 git checkout .
6 git init
7 ls
8 ls -al

```

Então temos a seguinte tela:

```

jmmr@jmmr-N435M: ~/web-project
jmmr@jmmr-N435M:~$ git clone https://github.com/masternobelphysics/web-project
Cloning into 'web-project'...
remote: Enumerating objects: 93, done.
remote: Counting objects: 100% (28/28), done.
remote: Compressing objects: 100% (19/19), done.
remote: Total 93 (delta 15), reused 9 (delta 9), pack-reused 65 (from 1)
Receiving objects: 100% (93/93), 297.63 KiB | 379.00 KiB/s, done.
Resolving deltas: 100% (27/27), done.
jmmr@jmmr-N435M:~$ cd web-project/
jmmr@jmmr-N435M:~/web-project$ pwd
/home/jmmr/web-project
jmmr@jmmr-N435M:~/web-project$ ls
404.html  browserconfig.xml  favicon.ico  index.html  test
apple-touch-icon.png  crossdomain.xml  fonts       js          title.png
arqteste.txt  css               humans.txt  robots.txt  title-wide.png
jmmr@jmmr-N435M:~/web-project$ git checkout .
Updated 0 paths from the index
jmmr@jmmr-N435M:~/web-project$ git init
Reinitialized existing Git repository in /home/jmmr/web-project/.git/
jmmr@jmmr-N435M:~/web-project$ ls
404.html  browserconfig.xml  favicon.ico  index.html  test
apple-touch-icon.png  crossdomain.xml  fonts       js          title.png
arqteste.txt  css               humans.txt  robots.txt  title-wide.png
jmmr@jmmr-N435M:~/web-project$ ls -al
total 116
drwxrwxr-x 6 jmmr jmmr 4096 ago 21 01:03 .
drwxr-x-- 19 jmmr jmmr 4096 ago 21 01:03 ..
-rw-rw-r-- 1 jmmr jmmr 1272 ago 21 01:03 404.html
-rw-rw-r-- 1 jmmr jmmr 3959 ago 21 01:03 apple-touch-icon.png
-rw-rw-r-- 1 jmmr jmmr 58 ago 21 01:03 arqteste.txt
-rw-rw-r-- 1 jmmr jmmr 416 ago 21 01:03 browserconfig.xml
-rw-rw-r-- 1 jmmr jmmr 603 ago 21 01:03 crossdomain.xml
drwxrwxr-x 2 jmmr jmmr 4096 ago 21 01:03 css
-rw-rw-r-- 1 jmmr jmmr 766 ago 21 01:03 favicon.ico
drwxrwxr-x 2 jmmr jmmr 4096 ago 21 01:03 fonts
drwxrwxr-x 8 jmmr jmmr 4096 ago 21 01:09 git
-rw-rw-r-- 1 jmmr jmmr 39014 ago 21 01:03 .htaccess
-rw-rw-r-- 1 jmmr jmmr 307 ago 21 01:03 humans.txt

```

Figura 4: Cópia do repositório, comandos de entrada dos arquivos e mostragem de datas de status da pasta.

Nesse caso, selecionamos o arquivo robots.txt no repositório no site do github para mudarmos algo nele, e acrescentamos texto nesse arquivo e clicamos commit changes para salvar as alterações:

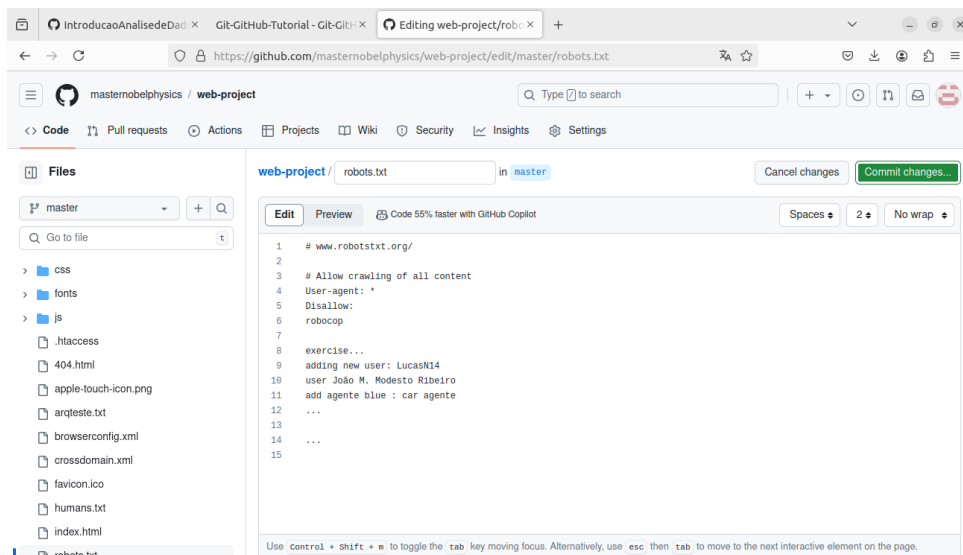


Figura 5: Alteração do arquivo robots.txt e o salvar da alteração.

Agora para este arquivo, vamos trabalhar o arquivo robots.txt do repositório dentro do terminal para o monitorar a versão mudada do site do github do mesmo saber do status, e para preparar o arquivo para o commit:

```

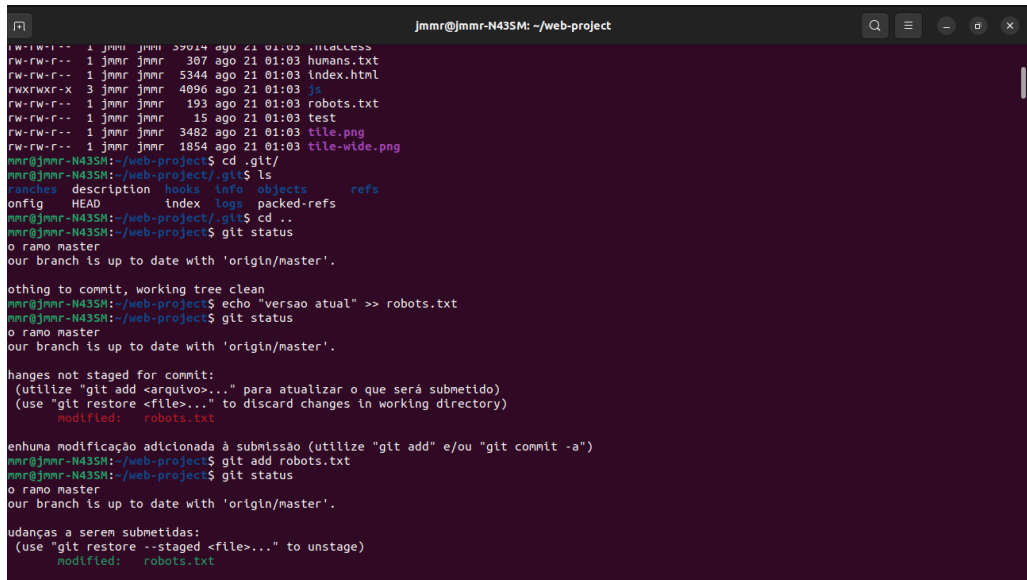
1 cd .git/
2 ls
3 cd ..
4 git status

```

Criamos um novo arquivo para versionar o atual de mesmo nome, começamos o monitoramento dele e ver o status dele:

```
1 echo "versao atual" >> robots.txt
2 git status
3 git add robots.txt
4 git status
```

Logo temos que a seguinte tela:



```
jmmr@jmmr-N43SM: ~/web-project
1 jmmr jmmr 307 ago 21 01:03 .htaccess
1 jmmr jmmr 5344 ago 21 01:03 index.html
3 jmmr jmmr 4096 ago 21 01:03 js
1 jmmr jmmr 193 ago 21 01:03 robots.txt
1 jmmr jmmr 15 ago 21 01:03 test
1 jmmr jmmr 3482 ago 21 01:03 tile.png
1 jmmr jmmr 1854 ago 21 01:03 tile-wide.png
jmmr@jmmr-N43SM: ~/web-project$ cd .git/
jmmr@jmmr-N43SM: ~/web-project/.git$ ls
branches  description  hooks  info  objects  refs
onfig  HEAD      index  logs  packed-refs
jmmr@jmmr-N43SM: ~/web-project/.git$ cd ..
jmmr@jmmr-N43SM: ~/web-project$ git status
o ramo master
our branch is up to date with 'origin/master'.

nothing to commit, working tree clean
jmmr@jmmr-N43SM: ~/web-project$ echo "versao atual" >> robots.txt
jmmr@jmmr-N43SM: ~/web-project$ git status
o ramo master
our branch is up to date with 'origin/master'.

changes not staged for commit:
  (utilize "git add <arquivo>..." para atualizar o que será submetido)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   robots.txt

nenhuma modificação adicionada à submissão (utilize "git add" e/ou "git commit -a")
jmmr@jmmr-N43SM: ~/web-project$ git add robots.txt
jmmr@jmmr-N43SM: ~/web-project$ git status
o ramo master
our branch is up to date with 'origin/master'.

alterações a serem submetidas:
  (use "git restore --staged <file>..." to unstage)
    modified:   robots.txt
```

Figura 6: Monitoramento e versionamento de arquivo.

Sendo assim com a preparação do commit changes, faremos o commit pelo terminal pela área da preparação do staging area dentro do repositório, e ver a nova alteração registrada nesse novo arquivo robots.txt:

```
1 git commit -m "simple modify"
2 git status
3 git log
```

Após isso, deveremos ter a seguinte tela, para a preparação da tela para o push:



```
jmmr@jmmr-N43SM: ~/web-project
jmmr@jmmr-N43SM: ~/web-project$ git commit -m "simple modify"
[master 5a5bc4c] simple modify
1 file changed, 1 insertion(+)
jmmr@jmmr-N43SM: ~/web-project$ git status
o ramo master
eu ramo está à frente de 'origin/master' por 1 submissão.
  (use "git push" to publish your local commits)

nothing to commit, working tree clean
jmmr@jmmr-N43SM: ~/web-project$ git log
commit 5a5bc4c4f1bb9ab1bffdab60792be174144ac92 (HEAD -> master)
author: masternobelphysics <masternobelphysics@gmail.com>
date:   Wed Aug 21 01:16:25 2024 -0300

    simple modify

commit 2af6844a019496728b602066f24138db051dd0ba (origin/master, origin/HEAD)
author: João M. Modesto Ribeiro <89819244+masternobelphysics@users.noreply.github.com>
date:   Wed Aug 21 00:27:12 2024 -0300

    Update robots.txt
```

Figura 7: Monitoramento e versionamento de arquivo.

Com isso, vamos fazer a transição para o repositório remoto para trabalhar o seu monitoramento pelo terminal:

```
1 git remote -v
2 git remote set-url https://github.com/masternobelphysics/web-project.git
3 git remote -v
```

```
jmmr@jmmr-N43SM: ~/web-project
jmmr@jmmr-N43SM:~/web-project$ git remote -v
origin https://github.com/masternobelphysics/web-project (fetch)
origin https://github.com/masternobelphysics/web-project (push)
jmmr@jmmr-N43SM:~/web-project$ git remote set-url origin https://github.com/masternobelphysics/web-project.git
jmmr@jmmr-N43SM:~/web-project$ git remote -v
origin https://github.com/masternobelphysics/web-project.git (fetch)
origin https://github.com/masternobelphysics/web-project.git (push)
```

Figura 8: Trabalhando com o repositório remoto.

Antes de dar o pull request, devemos criar uma branch que seria a caracterização da alteração feita no arquivo em track no robots.txt em forma do commit para ser uma mensagem da modificação, e o registro é dado mais recente nesse arquivo, logo:

```
1 git branch
2 git branch -v
3 git branch work
4 git checkout work
5 git log --all
```

```
jmmr@jmmr-N43SM:~/web-project$ git branch
* master
jmmr@jmmr-N43SM:~/web-project$ git branch -v
* master 5a5bc4c [ahead 1] simple modify
jmmr@jmmr-N43SM:~/web-project$ git branch work
jmmr@jmmr-N43SM:~/web-project$ git checkout work
Switched to branch 'work'
jmmr@jmmr-N43SM:~/web-project$ git log --all
commit 5a5bc4c4f1bb9ab1bffd9ab60792be174144ac92 (HEAD -> work, master)
Author: masternobelphysics <masterfisco@gmail.com>
Date: Wed Aug 21 01:16:25 2024 -0300

    simple modify

commit 905328aef1c4de571cdeae9f0202f807852f39e (origin/master, origin/patch-1)
Author: João M. Modesto Ribeiro <B9819244+masternobelphysics@users.noreply.github.com>
Date: Wed Aug 21 00:32:49 2024 -0300

    Update robots.txt

commit 2af6844a019496728b602066f24138db051dd0ba (origin/master, origin/HEAD)
Author: João M. Modesto Ribeiro <B9819244+masternobelphysics@users.noreply.github.com>
Date: Wed Aug 21 00:27:12 2024 -0300

    Update robots.txt
```

Figura 9: Branch.

Antes de digitar o comando `git push -u origin new_branch` no terminal, é necessário gerar o token que servirá como senha para que o comando funcione corretamente. Para isso, siga os passos abaixo no site do github para esta tela aqui abaixo, que no caso foi gerado um token para realização do trabalho, mas, iremos trabalhar esta tela:

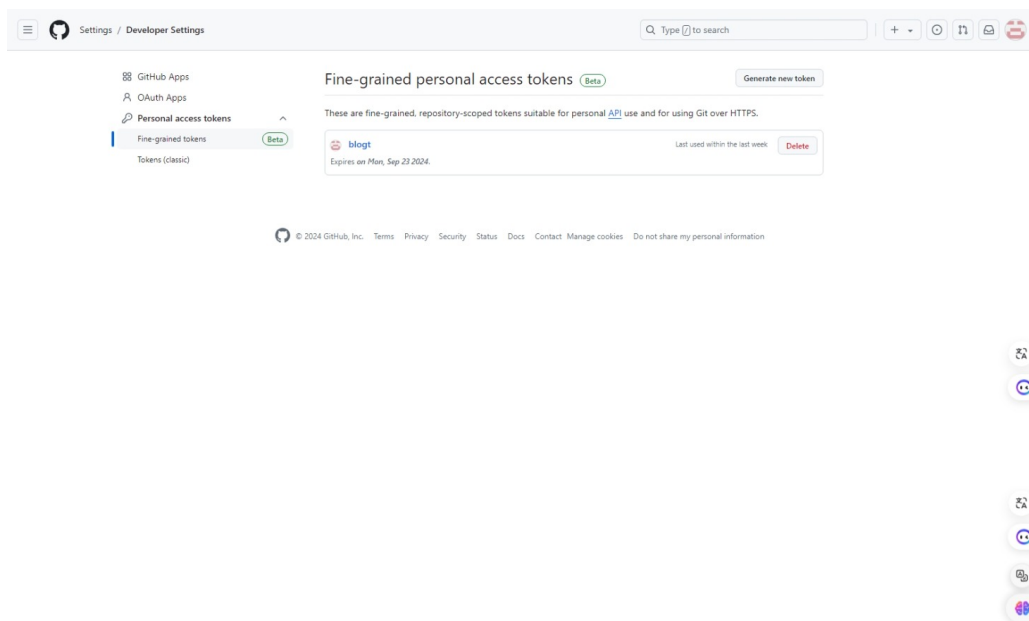


Figura 10: Tela de geração do token.

1. Clique na sua foto de perfil, depois em "Settings".
2. Clique em "Developer settings" e em seguida em "Personal access tokens".
3. Por fim, clique em "Fine-grained tokens". Para gerar o token, é preciso seguir estes passos:
4. Primeiro passo: Clique em "Generate new token".
5. Segundo passo: Dê um nome ao token ("Token name") e escolha a data de expiração na opção "Expiration".
6. Terceiro passo: Em "Repository access", selecione "All repositories". Em "Repository permissions", defina "Read and write" para todas as opções disponíveis e "Read" se essa for a única opção.
7. Quarto passo: Clique em "Generate token" no final da página.
8. Quinto passo: Clique no ícone de duas folhas que fica ao lado dele para copiar o token gerado para colocar no comando do git push.

Após isso, execute o comando `git push -u origin new_branch`, após passos anteriores, onde pedirá o username e a senha pedida pelo terminal será este token gerado, logo teremos esta tela abaixo em que criou o pull request a ser dado no site do github:

```
jmmr@jmmr-N435M:~/web-project$ git remote set-url origin https://github.com/masternobelphysics/web-project.git
jmmr@jmmr-N435M:~/web-project$ git status
No ramo new_branch
nothing to commit, working tree clean
jmmr@jmmr-N435M:~/web-project$ git branch
* master
  new_branch
  work
jmmr@jmmr-N435M:~/web-project$ ls
404.html      arqteste.txt      crossdomain.xml  favicon.ico       humans.txt       js               test            tile-wide.png
apple-touch-icon.png  browserconfig.xml  css              fonts             index.html       robots.txt      tile.png
jmmr@jmmr-N435M:~/web-project$ git push -u origin new_branch
Username for 'https://github.com': masternobelphysics
Password for 'https://github.com': 
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 304 bytes | 304.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
remote: Create a pull request for 'new_branch' on GitHub by visiting:
remote:   https://github.com/masternobelphysics/web-project/pull/new/new_branch
remote:
To https://github.com/masternobelphysics/web-project.git
 * [new branch]      new_branch -> new_branch
Branch 'new_branch' set up to track remote branch 'new_branch' from 'origin'.
jmmr@jmmr-N435M:~/web-project$
```

Figura 11: Uso dos comandos do git e git push até a criação do pull request através do branch.

Agora vamos ao open pull request do repositório, em que enviaremos a alteração para análise e aceitação para o membro do repositório original através da origem do fork do repositório original até o envio do branch para que a pessoa dona do repositório aceite ou não alteração através do pull request, com isso temos a seguinte tela do github abaixo:

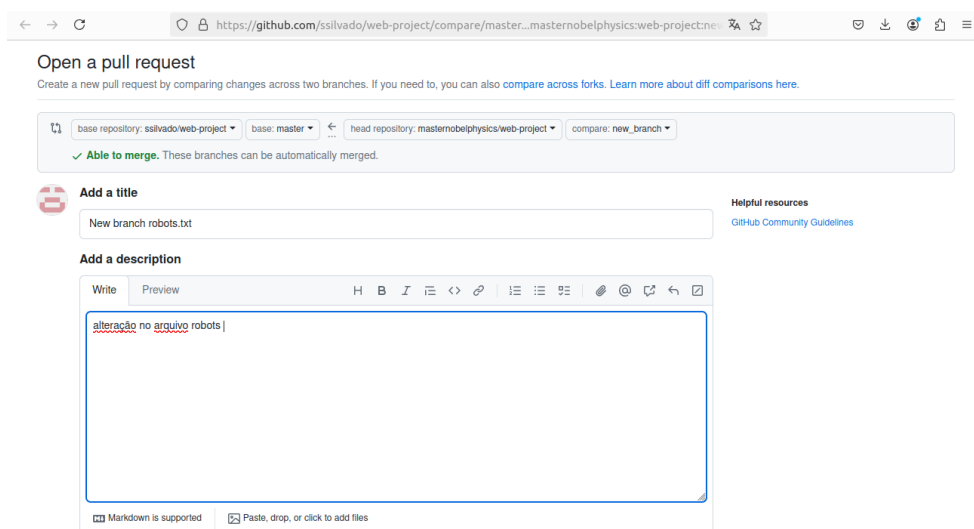


Figura 12: Tela do open pull request.

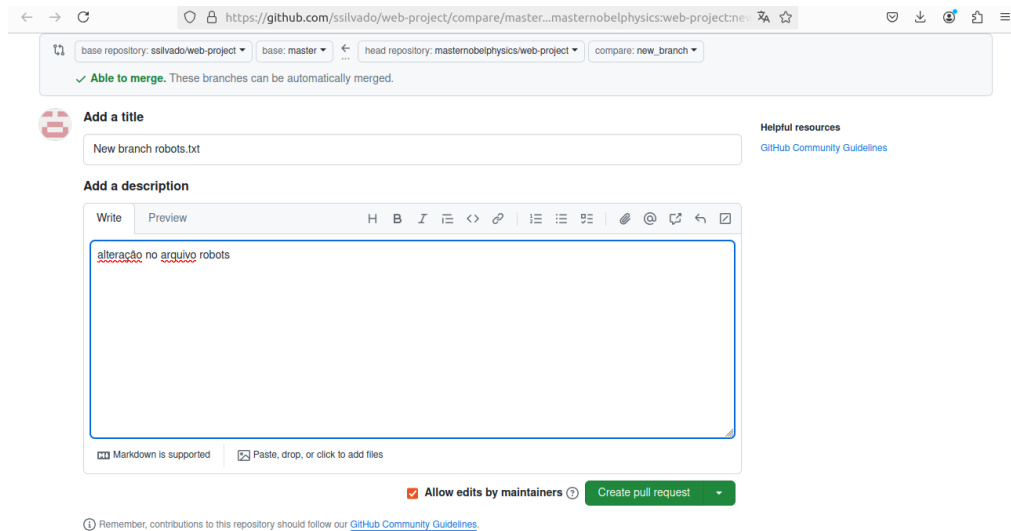


Figura 13: Tela da localidade para a execução do ícone verde do pull request na mesma tela.

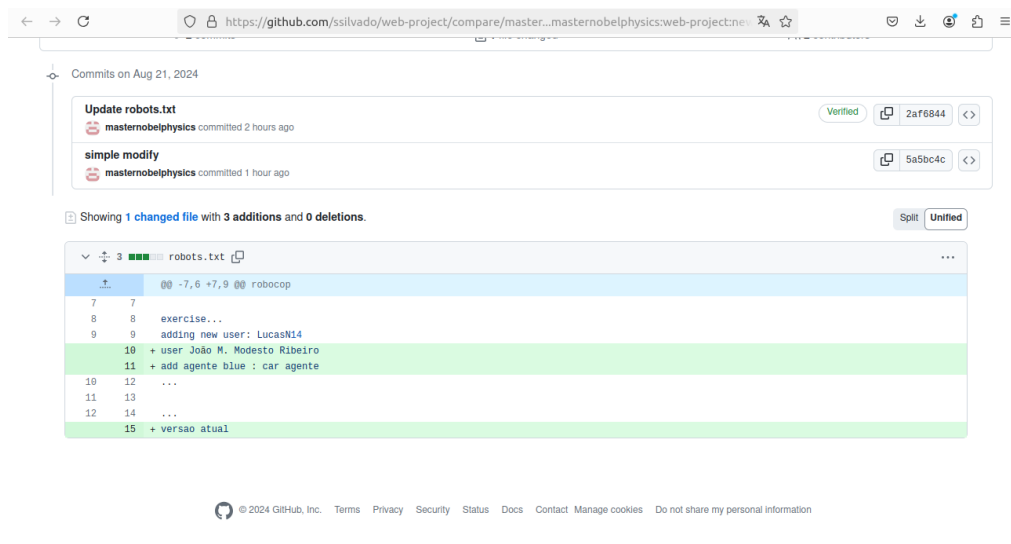


Figura 14: Amostra da alteração para ser aceita ou não antes do pull request na mesma tela.

Após isso, temos a finalização do pull request, dado pela New branch 'robots.txt' em status closed na figura abaixo:

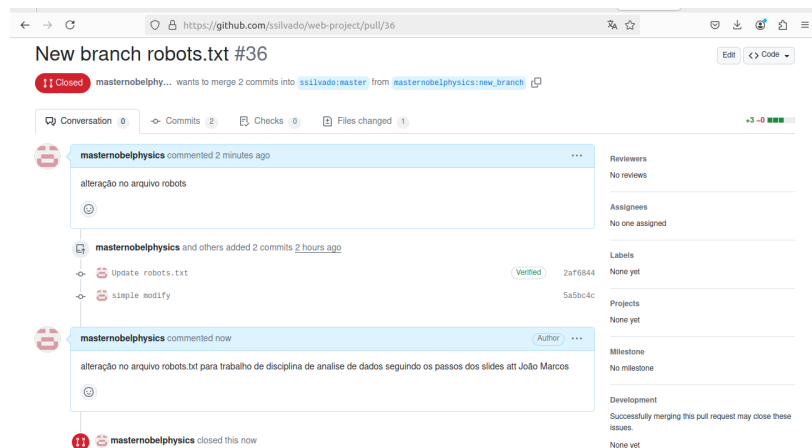


Figura 15: New branch closed enviado pelo pull request para aceitação ou não do usuário dono do repositório original.