

SESSÃO 4

Primeiros Passos com Git

Objetivos:

- Recuar no tempo
 - posicionamento num commit específico
 - identificar problemas
 - recuperar ficheiros eliminados
 - descartar commits
- Criar e gerir tags
- Exportação de repositórios

Transcrição

Vídeo 8

<https://www.youtube.com/watch?v=MLztwr8J9H4>

Nota: antes de realizar cada uma das atividades propostas no vídeo é necessário que não haja alterações para incluir em *commits*

1. Adicionar o repositório Site ZX Spectrum

- a) menu **File** comando **Add local repository ...**

2. Para verificar como estava o projeto quando foi efetuado o segundo *commit* [Adicionar conteúdo]

- a) abrir a linha de comando (**botão direito do rato** > **Abrir no terminal** > seta v > **Git Bash**)
- b) ir para a pasta do repositório

```
cd "C:\Users\ls\Desktop\Site ZX Spectrum"
```

- c) copiar o **hash** do segundo *commit*

selecionar o segundo *commit* e na parte superior, a seguir aos números, clicar nos dois quadrados para copiar



- d) aplicar o comando **git checkout** ao **hash** do segundo *commit*

```
MINGW64:sl
sl@pc MINGW64 ~
$ cd "C:\Users\sl\Desktop\Site ZX Spectrum"

sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$ git checkout 9390e6b1122c39f7aefb89cb31a2f3ba9d43d0a4
Note: switching to '9390e6b1122c39f7aefb89cb31a2f3ba9d43d0a4'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at 9390e6b Adicionar conteúdo

sl@pc MINGW64 ~/Desktop/Site ZX Spectrum ((9390e6b...))
$
```

Nota: Ao verificar o site este aparece com o conteúdo que tinha aquando do segundo *commit*

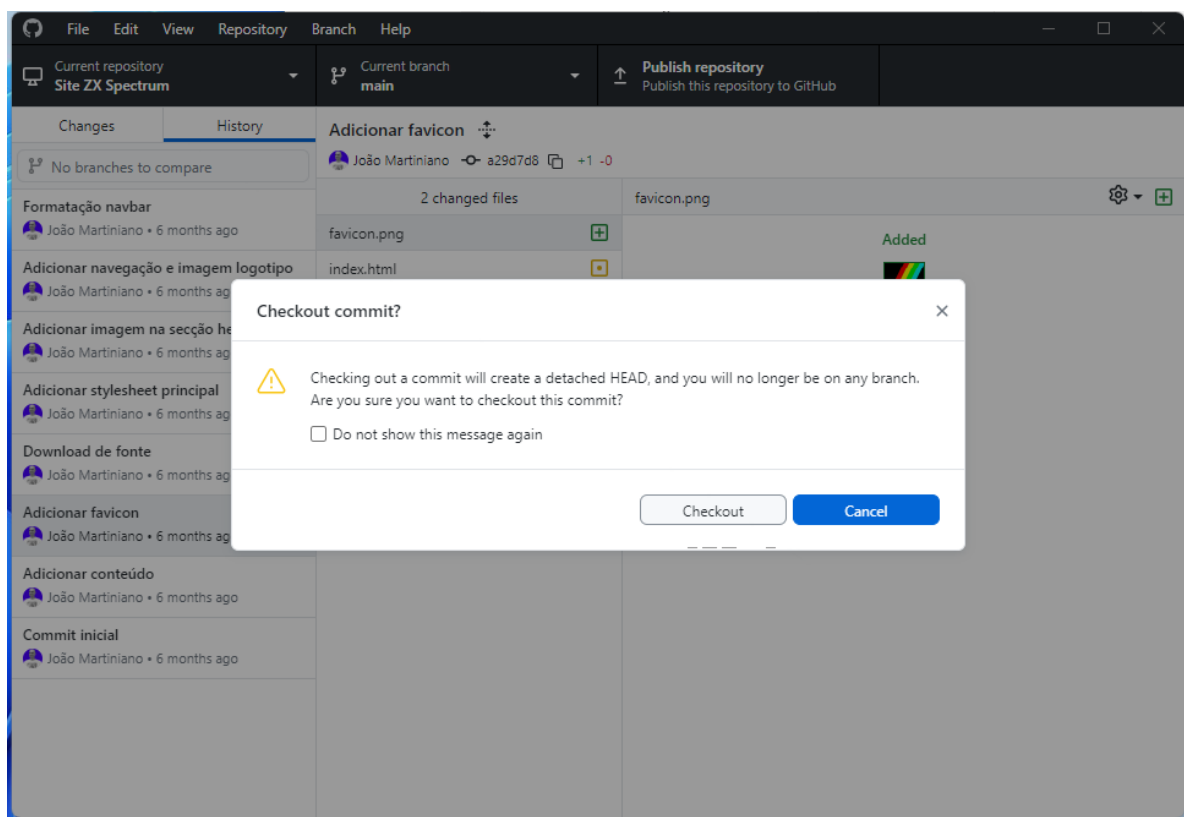
e) Para **reverter** o passo anterior executar **git checkout main**

git checkout main

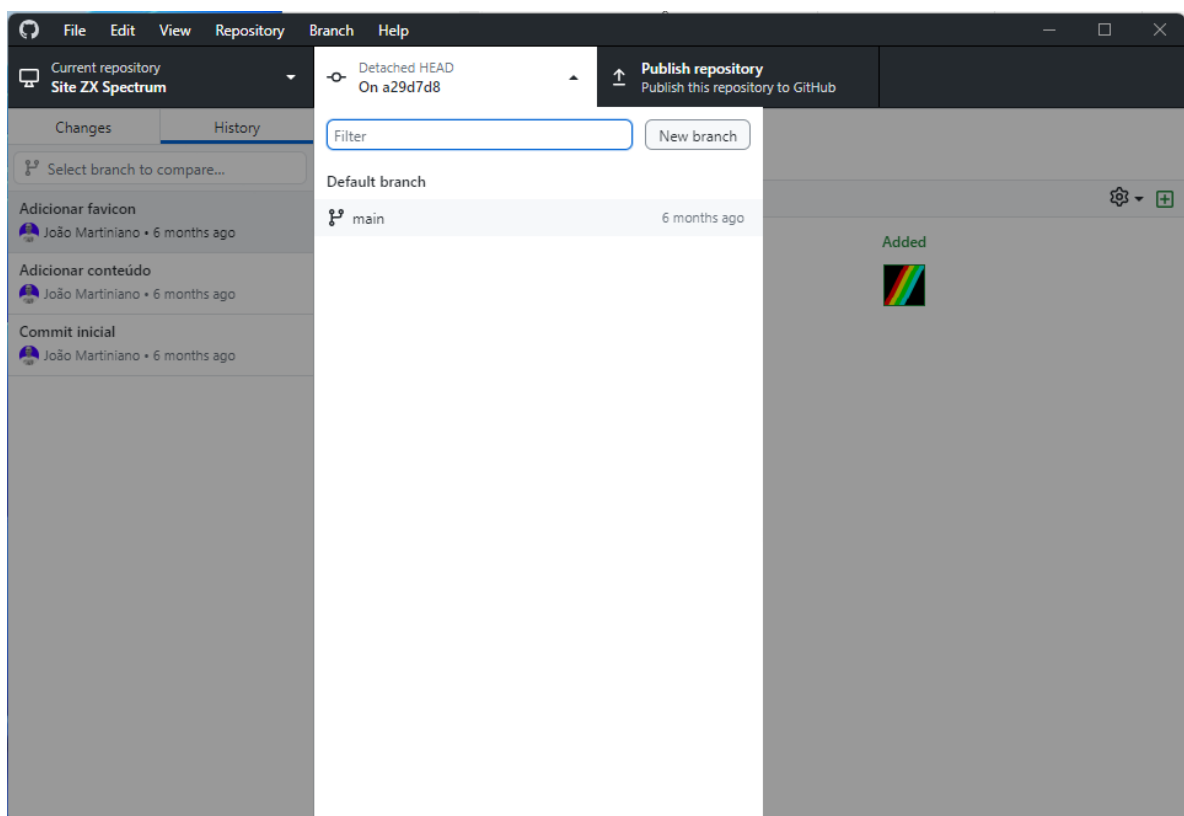
```
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum ((9390e6b...))
$ git checkout main
Previous HEAD position was 9390e6b Adicionar conteúdo
Switched to branch 'main'

sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$
```

f) No **Git Desktop** clicar com o botão direito do rato sobre o *commit* para onde se pretende voltar e no menu aberto clicar em *checkout commit* e na janela aberta clicar *checkout*.

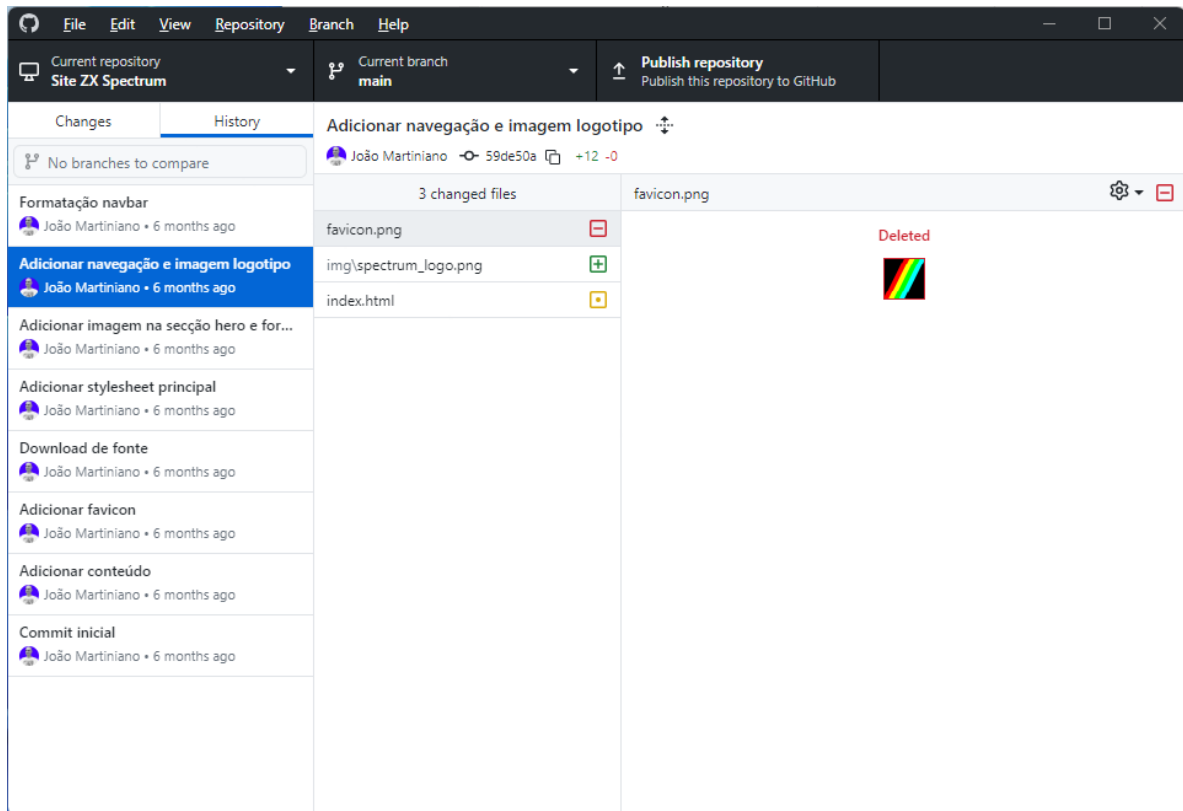


g) Para voltar a ver todos os **commits** em **Detach HEAD** selecionar **main**



3. Recuperar um ficheiro eliminado

Método 1 - analisar através do Git Desktop o que aconteceu ao ficheiro



Nota: O ficheiro neste **commit** deixa de existir.

Método 2 - analisar através da linha de comandos

```
MINGW64:~$ cd ~/Desktop/Site_ZX_Spectrum
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$ git log -- favicon.png
commit 59de50af6a62b908cc78b9fa3bbd3a58dc2339ca
Author: João Martiniano <44736477+joaomartiniano@users.noreply.github.com>
Date: Mon Sep 4 15:15:23 2023 +0100

    Adicionar navegação e imagem logotipo

commit a29d7d8fa1b3d9b355c5d832e45632010d628665
Author: João Martiniano <44736477+joaomartiniano@users.noreply.github.com>
Date: Mon Sep 4 14:40:53 2023 +0100

    Adicionar favicon

sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$ git show 59de50af6a62b908cc78b9fa3bbd3a58dc2339ca
```

```
MINGW64:sl
commit 59de50af6a62b908cc78b9fa3bbd3a58dc2339ca
Author: João Martiniano <44736477+joaomartiniano@users.noreply.github.com>
Date: Mon Sep 4 15:15:23 2023 +0100

    Adicionar navegação e imagem logotipo

diff --git a/favicon.png b/favicon.png
deleted file mode 100644
index e5fc9f3..0000000
Binary files a/favicon.png and /dev/null differ
diff --git a/img/spectrum_logo.png b/img/spectrum_logo.png
new file mode 100644
index 0000000..ccb14f8
Binary files /dev/null and b/img/spectrum_logo.png differ
diff --git a/index.html b/index.html
index be4fd75..935247a 100644
--- a/index.html
:|
```

a) Fazer **checkout** no último **commit** que o **favicon.png** estava presente

```
MINGW64:sl
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$ git checkout f41a74d40b96591642f7d296b5fd6c16a3f86395
Note: switching to 'f41a74d40b96591642f7d296b5fd6c16a3f86395'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

    git switch -c <new-branch-name>

Or undo this operation with:

    git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at f41a74d Adicionar imagem na secção hero e formatar secções
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum ((f41a74d...))
$ |
```

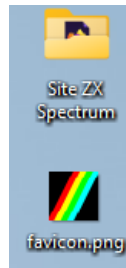
- b) Agora o ficheiro está na pasta.
- c) Passar o ficheiro para o ambiente de trabalho.



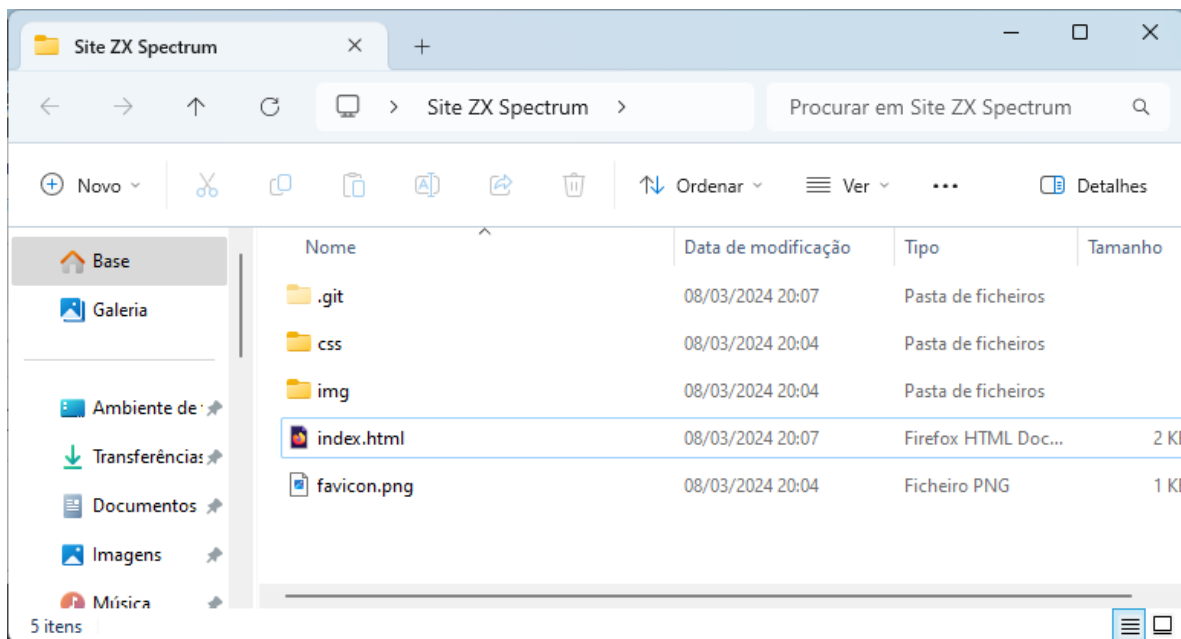
d) Fazer **checkout** para o **main** [voltar ao último estado]

```
MINGW64:sl
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum ((f41a74d...))
$ git checkout main
Previous HEAD position was f41a74d Adicionar imagem na secção hero e formatar secções
Switched to branch 'main'

sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$
```



e) Copiar o ficheiro do ambiente de trabalho para a pasta do projeto



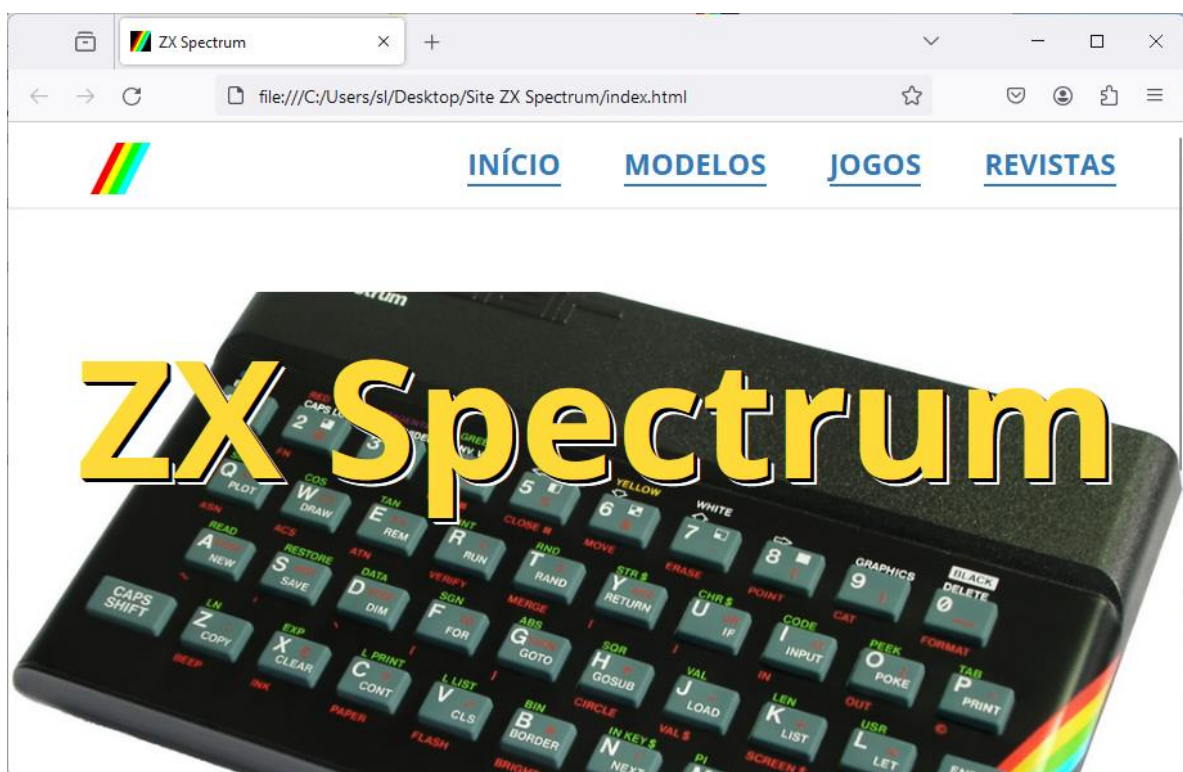
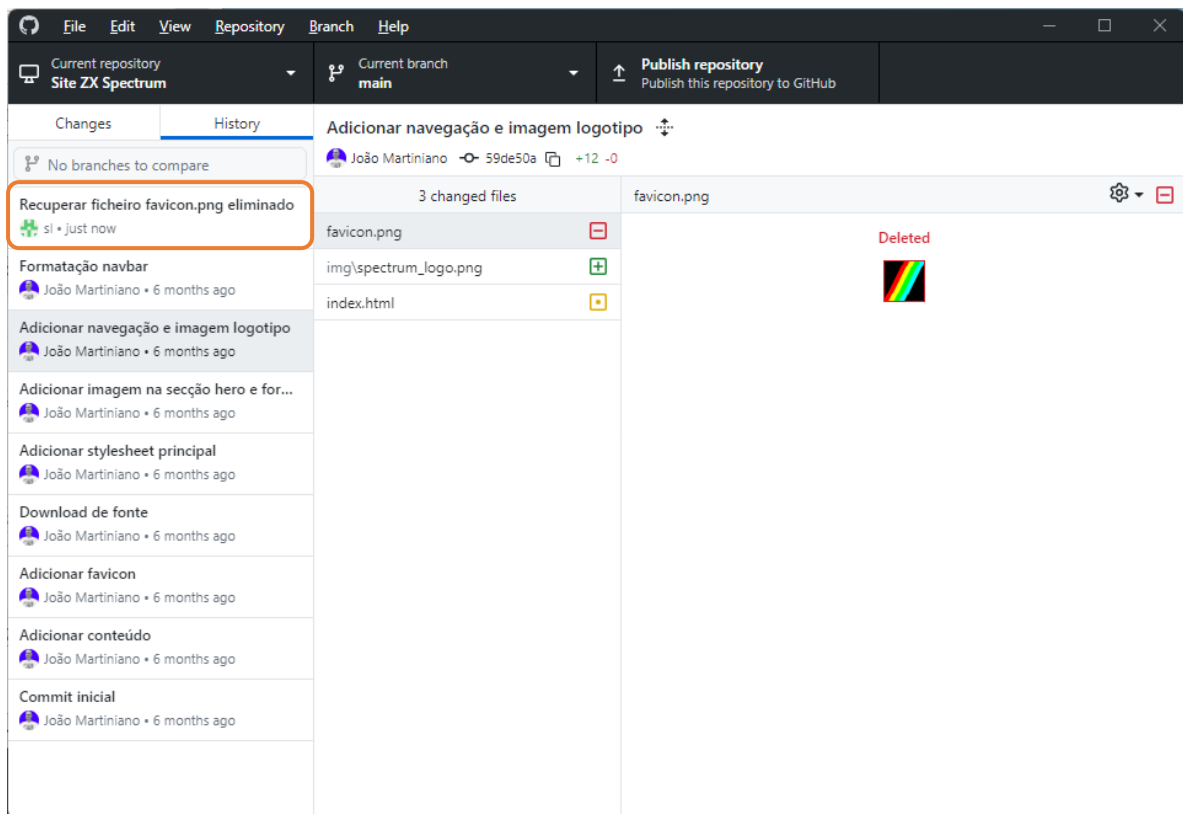
f) Ou através da linha de comandos usar o comando seguinte para recuperar o ficheiro.

git checkout [hash do commit]^ [nome do ficheiro a recuperar]

```
MINGW64:sl
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$ git checkout 59de50af6a62b908cc78b9fa3bbd3a58dc2339ca^ favicon.png
Updated 1 path from 7af8ecd

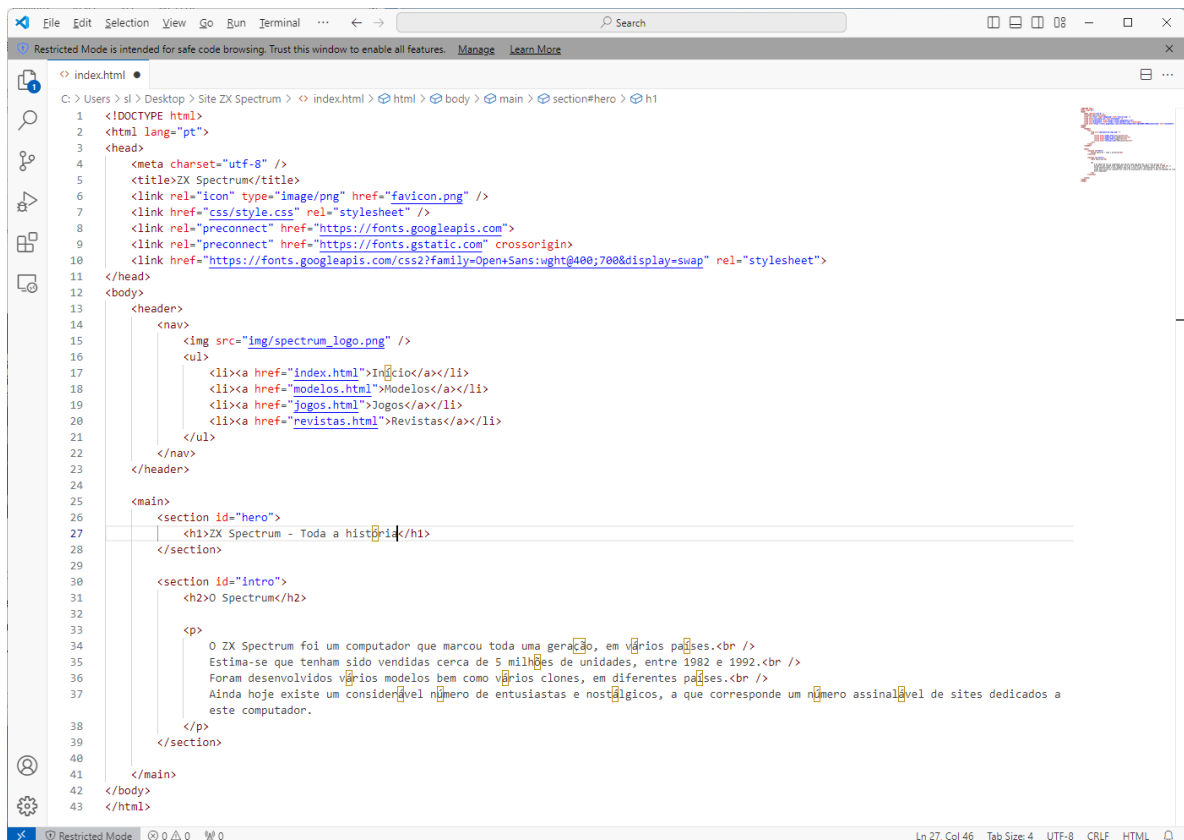
sl@pc MINGW64 ~/Desktop/Site ZX Spectrum (main)
$
```

4. Escrever *commit*

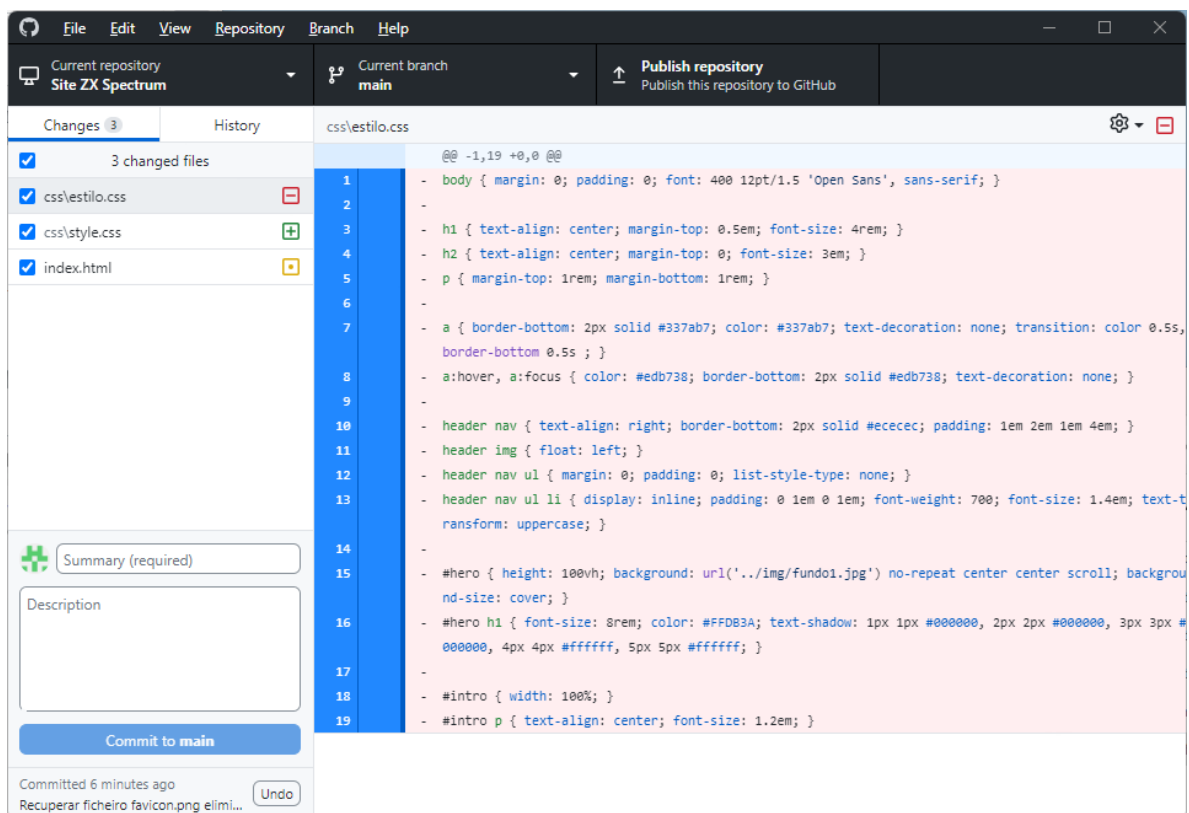


5. Fazer as seguintes alterações no site:

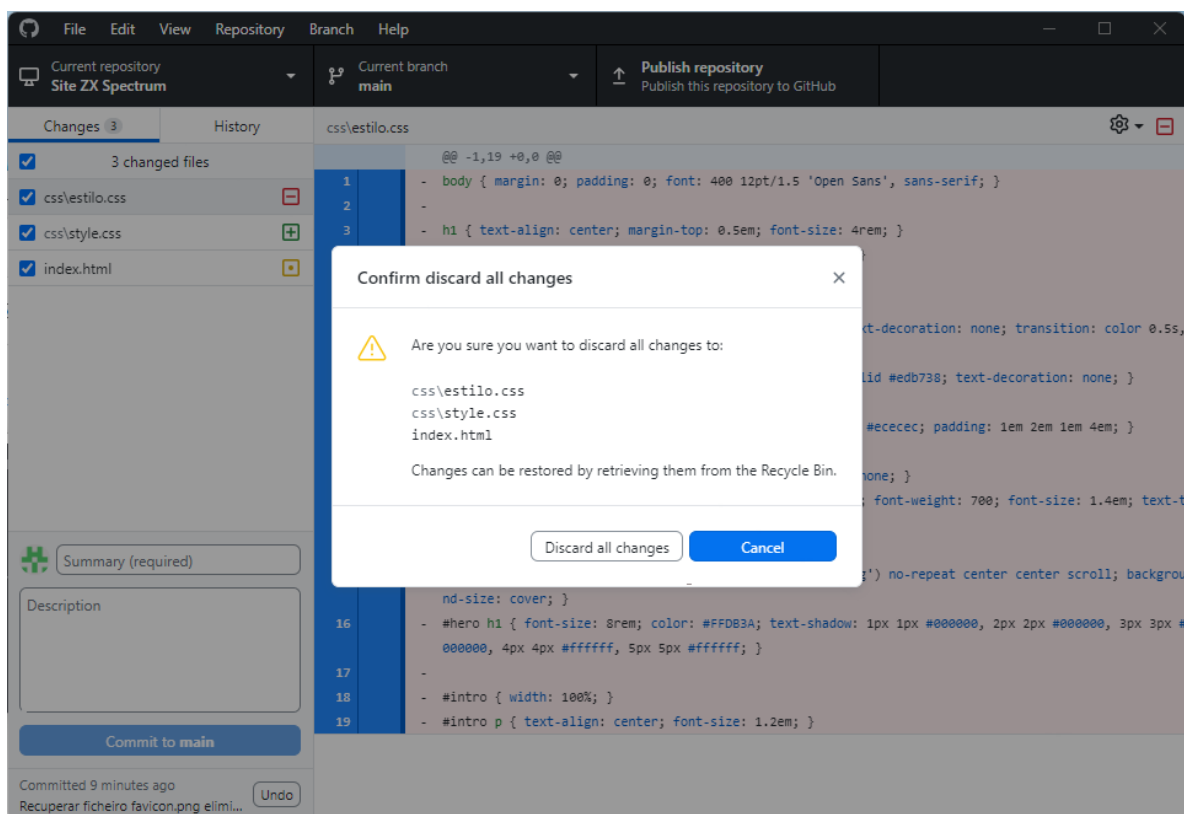
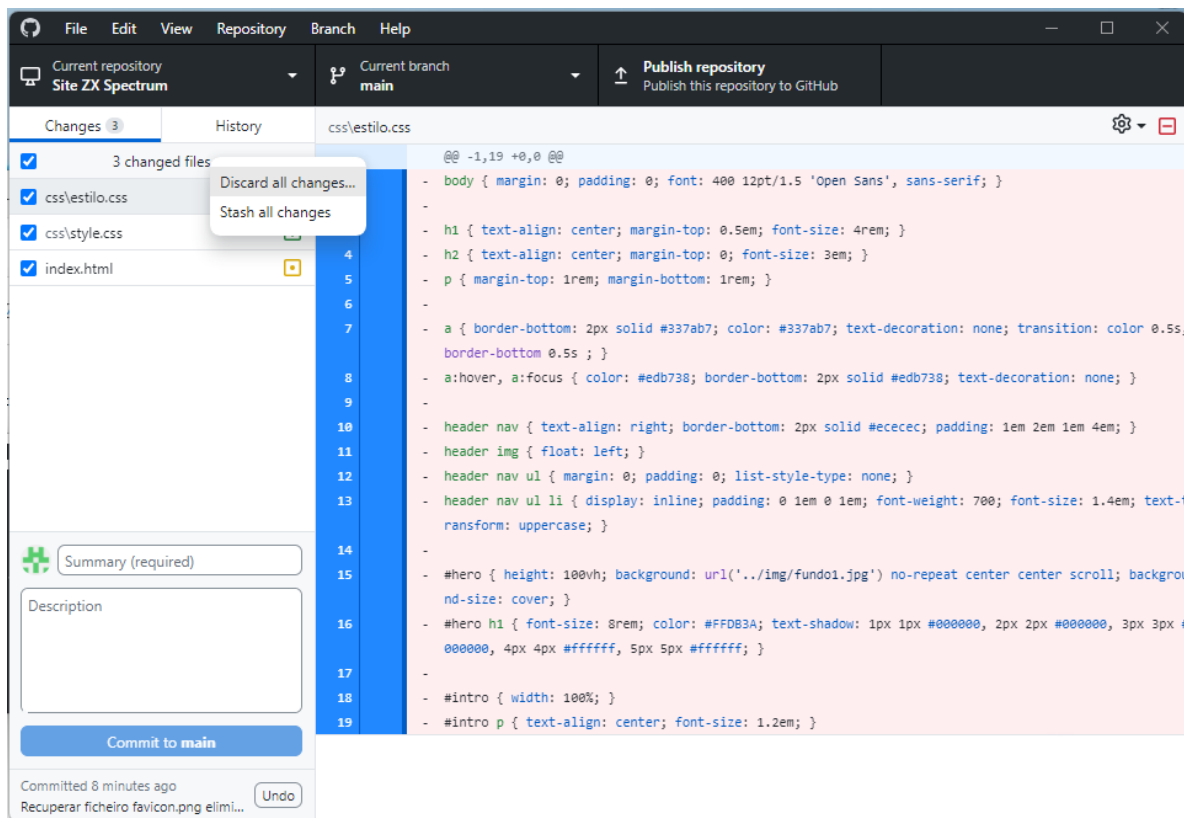
- alterar o nome do ficheiro de estilos para style.css
- alterar no index.html o ficheiro de estilo
- alterar o h1 para ZX Spectrum – Toda a história



```
1 <!DOCTYPE html>
2 <html lang="pt">
3 <head>
4   <meta charset="utf-8" />
5   <title>ZX Spectrum</title>
6   <link rel="icon" type="image/png" href="favicon.png" />
7   <link href="css/style.css" rel="stylesheet" />
8   <link rel="preconnect" href="https://fonts.googleapis.com">
9   <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
10  <link href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@400;700&display=swap" rel="stylesheet">
11 </head>
12 <body>
13   <header>
14     <nav>
15       
16       <ul>
17         <li><a href="index.html">Início</a></li>
18         <li><a href="modelos.html">Modelos</a></li>
19         <li><a href="jogos.html">Jogos</a></li>
20         <li><a href="revistas.html">Revistas</a></li>
21       </ul>
22     </nav>
23   </header>
24
25   <main>
26     <section id="hero">
27       <h1>ZX Spectrum - Toda a história</h1>
28     </section>
29
30     <section id="intro">
31       <h2>O Spectrum</h2>
32
33       <p>
34         O ZX Spectrum foi um computador que marcou toda uma geração, em vários países.<br />
35         Estima-se que tenham sido vendidas cerca de 5 milhões de unidades, entre 1982 e 1992.<br />
36         Foram desenvolvidos vários modelos bem como vários clones, em diferentes países.<br />
37         Ainda hoje existe um considerável número de entusiastas e nostálgicos, a que corresponde um número assinalável de sites dedicados a este computador.
38       </p>
39     </section>
40
41   </main>
42 </body>
43 </html>
```



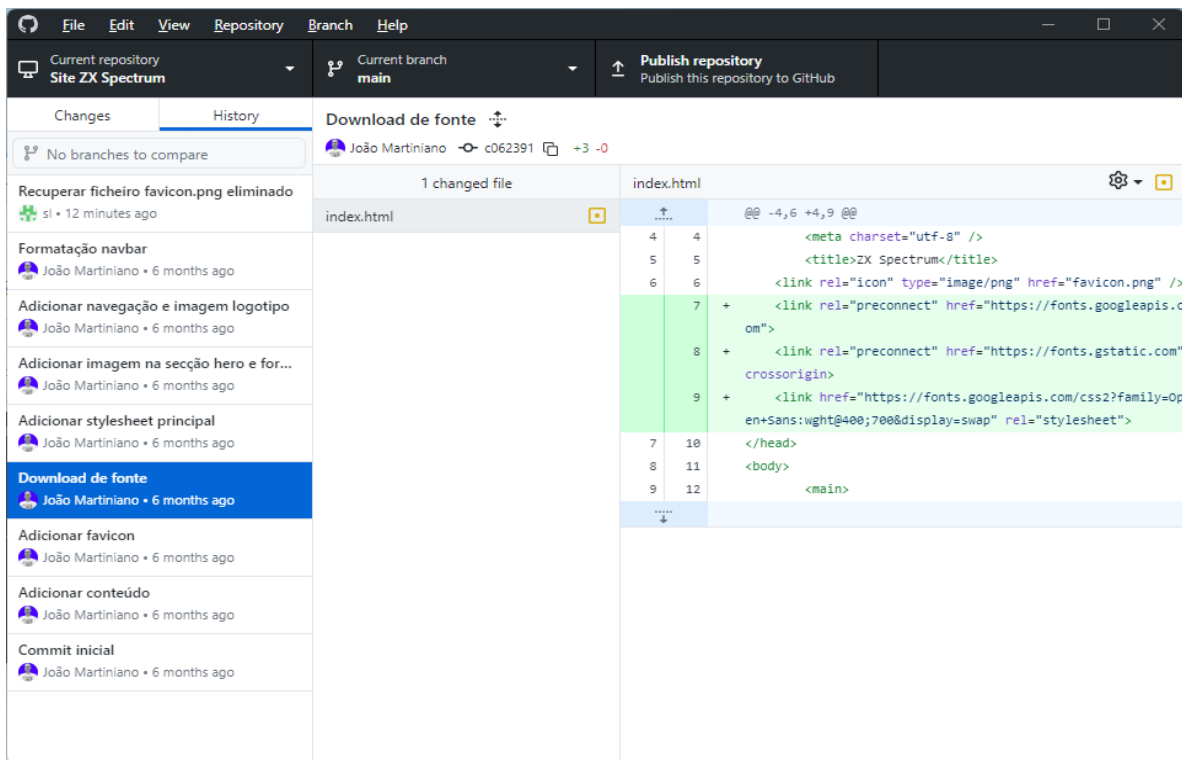
6. Para não aplicar as alterações que ainda não foram alvo de um *commit* seleccionar as alterações, clicar com o botão direito do rato sobre as mesmas e no menu de atalho **Discard all changes...**



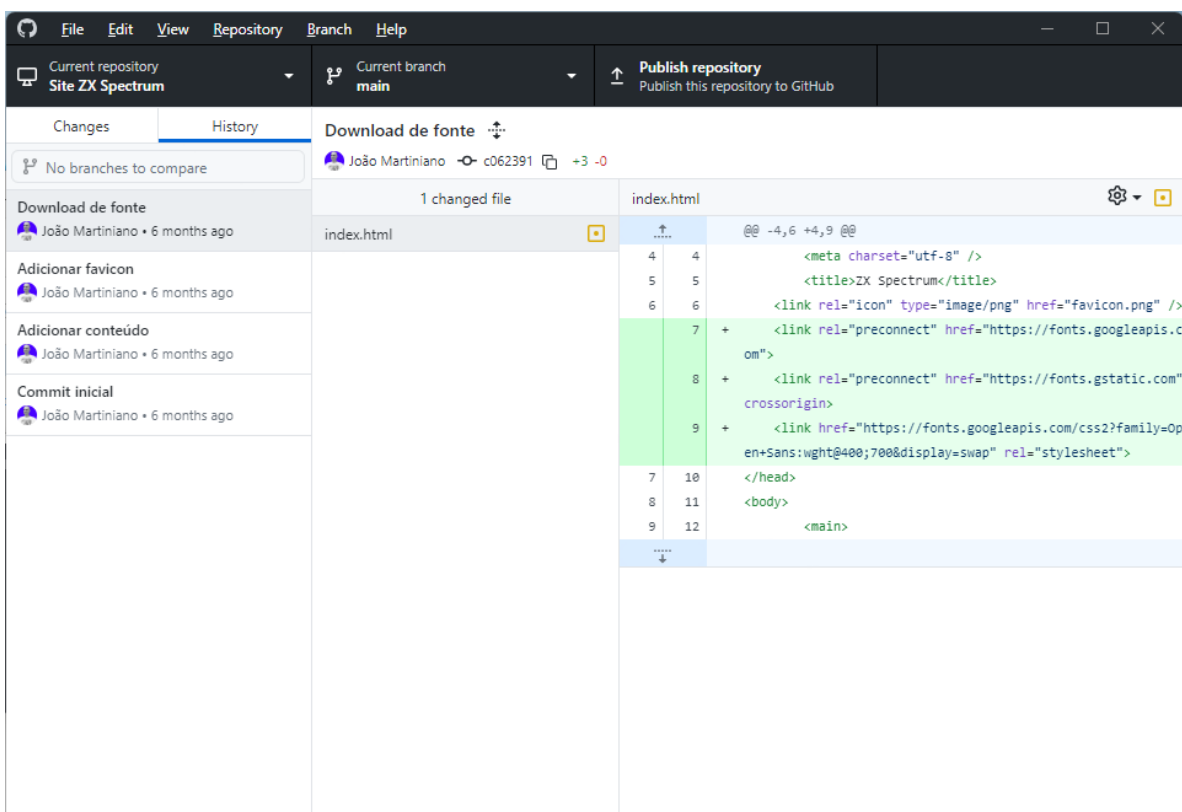
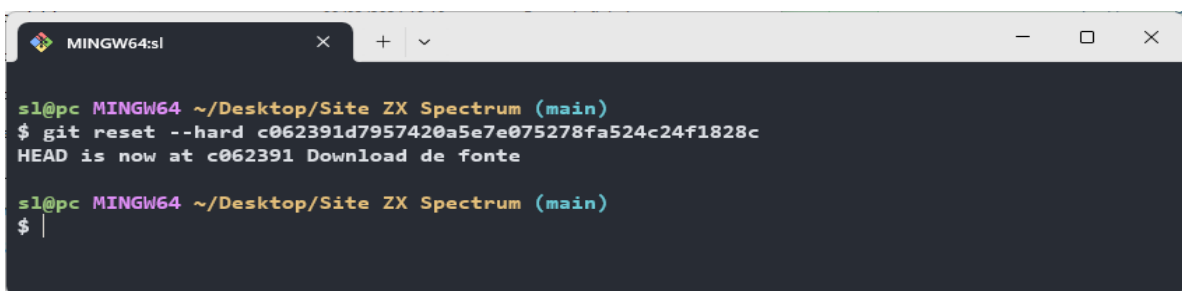
Nota: Os ficheiros afetados são enviados para a reciclagem do computador.

7. Voltar a um *commit* anterior

Nota: Esta operação não deve ser executada em repositórios partilhados com outras pessoas



8. Copiar o *hash* do *commit* e digitar o comando **git reset – hard** seguido do hash



<https://www.youtube.com/watch?v=1bTnNvQNd2Q>

Tag – permite associar um *commit* a um identificador; as tags permitem sinalizar um determinado momento num determinado ponto do desenvolvimento do projeto onde ocorreu algo especial

9. Adicionar o repositório calculos

a) menu **File** comando **Add local repository ...**

10. Criar uma tag usando a linha de comandos

```
MINGW64:sl
s1@pc MINGW64 ~
$ cd "C:\Users\s1\Desktop\repositório-calculos\calculos"

s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
$ git tag v0.8 a97e0ade9e4f316b1335de72dee906c93cf7f25d

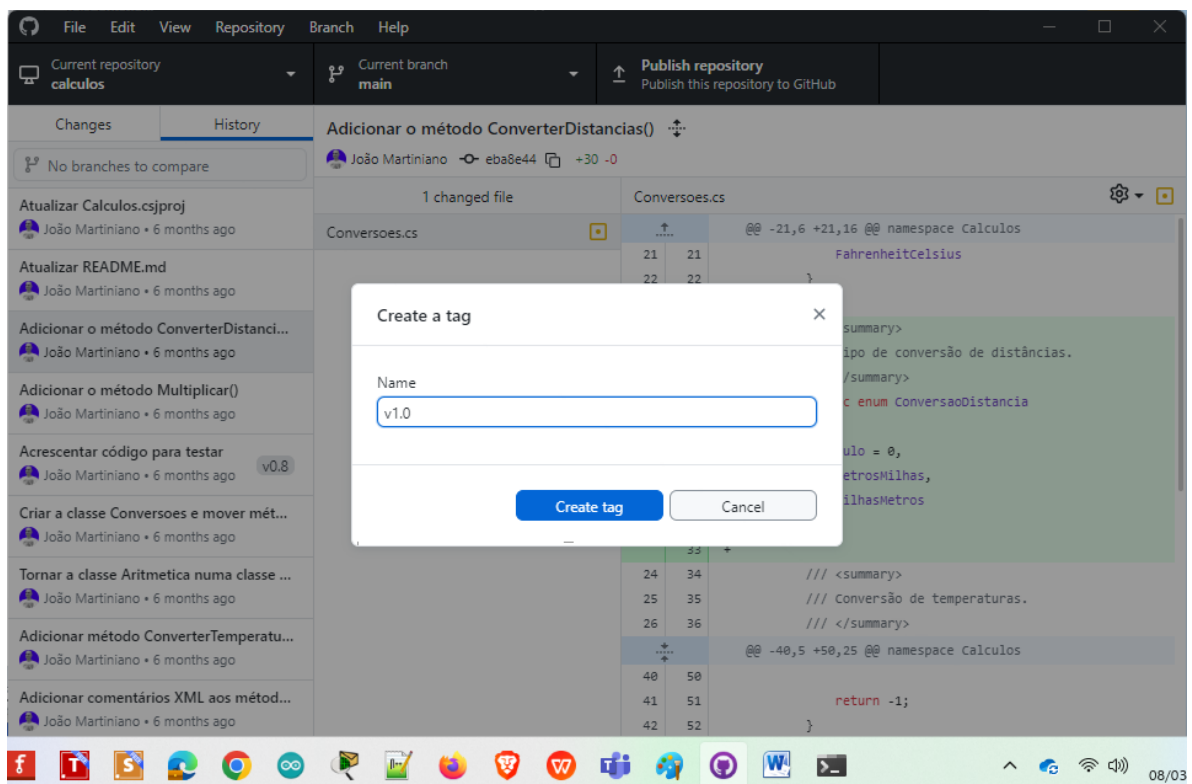
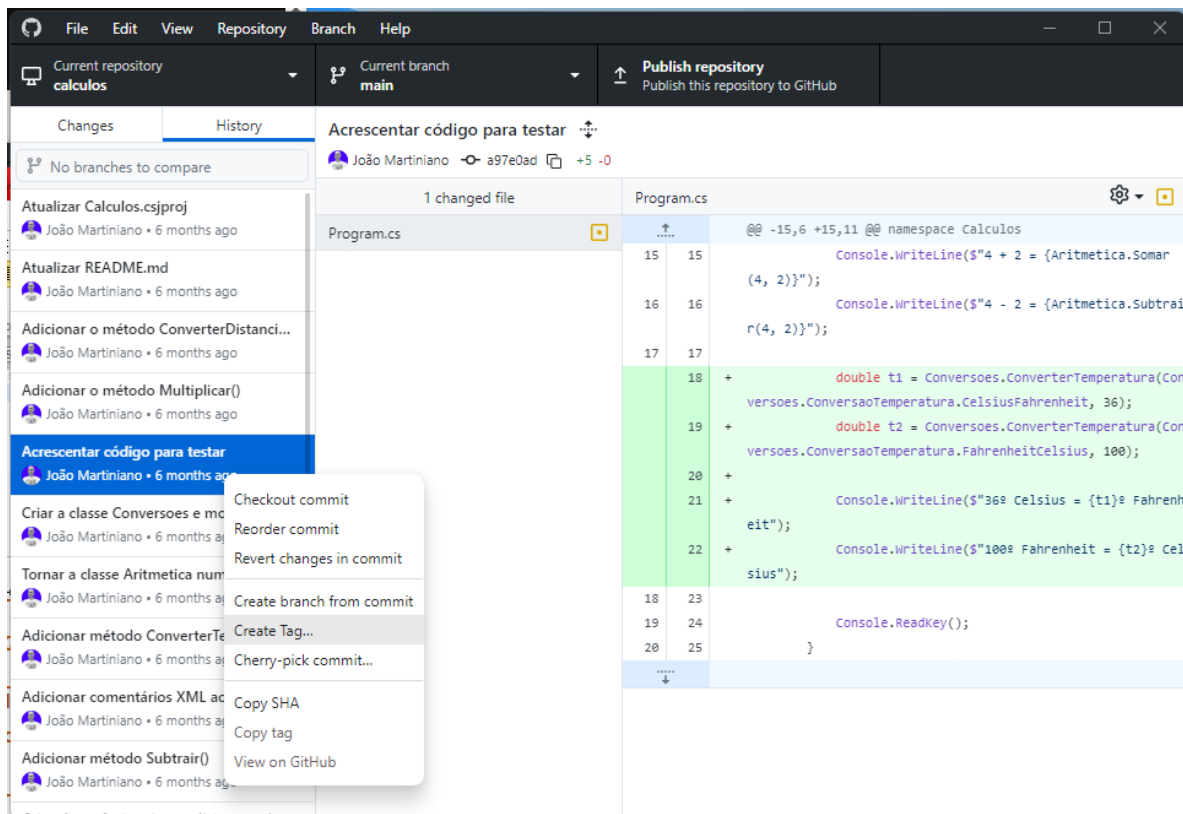
s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
$
```

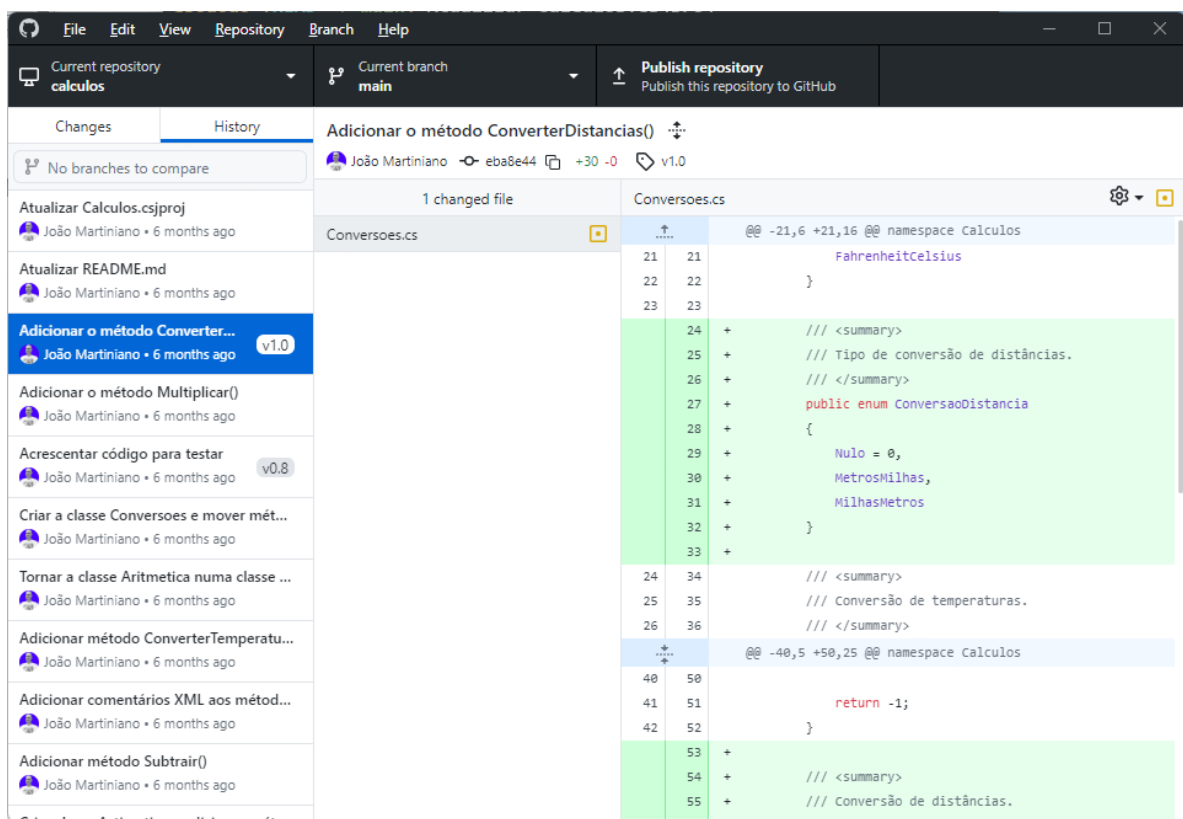
```
MINGW64:sl
s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
$ git tag
v0.8

s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
$ git log --oneline
eb6a6d0 (HEAD -> main) Atualizar Calculos.csproj
db1afa0 Atualizar README.md
eba8e44 Adicionar o método ConverterDistancias()
5e1076b Adicionar o método Multiplicar()
a97e0ad (tag: v0.8) Acrescentar código para testar
dc51200 Criar a classe Conversoes e mover método
f64f291 Tornar a classe Aritmetica numa classe estática
a3e8403 Adicionar método ConverterTemperatura()
a6adf3b Adicionar comentários XML aos métodos Somar() e Subtrair()
50c6d6f Adicionar método Subtrair()
45b8660 Criar classe Artimetica e adicionar método Somar()
3710295 Criar projeto
ce7971c Initial commit
```

11. Criar uma tag usando o *Git desktop*:

- a) clicar com o **botão direito do rato** sobre o **commit** *Acrescentar código para testar* e no **menu de atalho** selecionar **Create Tag...**





12. Para apagar uma tag pode utilizar um de dois processos

- clicar com o **botão direito do rato sobre o commit** *Acrescentar código para testar* e no **menu de atalho** selecionar **Delete Tag...**
- para **apagar tags mais antigas** na linha de comandos digitar

git tag -d v0.8

13. Para **atualizar o repositório remoto** depois da eliminação das **tags** clicar em **Push origin** para adicionar a **tag** readicionada pelo **Git Desktop**.

14. Para **atualizar o repositório remoto** depois da eliminação das **tags** pela linha de comandos digitar na linha de comandos:

git push origin v0.8

15. Exportar o repositório

- através do comando **git archive** exporte o **repositório** a partir da **v1.0**

git log --oneline

git log -- oneline	Para obter o <i>hash</i> condensado do <i>commit</i>
git archive-- format=zip --output=calculos-v1.0.zip <i>hash do commit</i>	Comando para exportar v1.0

```
MINGW64:sl
s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
$ git log --oneline
eb6a6d0 (HEAD -> main) Atualizar Calculos.csproj
db1afa0 Atualizar README.md
eba8e44 (tag: v1.0) Adicionar o método ConverterDistancias()
5e1076b Adicionar o método Multiplicar()
a97e0ad (tag: v0.8) Acrescentar código para testar
dc51200 Criar a classe Conversoes e mover método
f64f291 Tornar a classe Aritmetica numa classe estática
a3e8403 Adicionar método ConverterTemperatura()
a6adf3b Adicionar comentários XML aos métodos Somar() e Subtrair()
50c6d6f Adicionar método Subtrair()
45b8660 Criar classe Artimetica e adicionar método Somar()
3710295 Criar projeto
ce7971c Initial commit

s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
$ git archive --format=zip --output=calculos-v1.0.zip eba8e44

s1@pc MINGW64 ~/Desktop/repositório-calculos/calculos (main)
```

