

Versionamento com Git

Semana Nacional de T.I. - Laboratório de Transformação Digital

André Eppinghaus

André Eppinghaus

- Mestre em Avaliação pela Fundação Cesgranrio (2011)
- Pós-graduado em Inteligência Artificial e Aprendizado de Máquina PUC-Minas (2023)
- Graduação em Tecnologia em Processamento de Dados pela Universidade Castelo Branco (1998).
- Atualmente sou docente da Universidade Estácio de Sá nos cursos de Sistemas de Informação e Jogos Digitais.
- Analista de sistemas do Centro Nacional de Conservação da Flora na Diretoria de Pesquisas do Jardim Botânico no Rio de Janeiro.
- Autor da ferramenta de avaliação online CheckTask da empresa Ambiente Tecnológico LTDA. (2011)
- Responsável pelo desenvolvimento do software de avaliação da Universidade Gama Filho no período de 2006 a 2010.

Laboratório de Transformação Digital - LTD - 2024.1

O que é git ?

- Git é um sistema de controle de versão (VCS) distribuído
- [versionamento; versionar] É uma ferramenta útil para rastrear facilmente alterações em seu código, colaborar e compartilhar.

Pré-requisitos

- Para seguir o fluxo de trabalho do Git, você precisa de uma conta no GitHub e um

Termos do GitHub

Repositórios

- [histórico de revisões] Um repositório é onde o trabalho do seu projeto acontece; pense nele como a pasta do seu projeto. Ele contém todos os arquivos e o histórico de revisões do seu projeto.
- [colaboração] Você pode trabalhar sozinho em um repositório ou convidar outras pessoas para colaborar com você nesses arquivos.

Termos do GitHub

Clonagem (git clone)

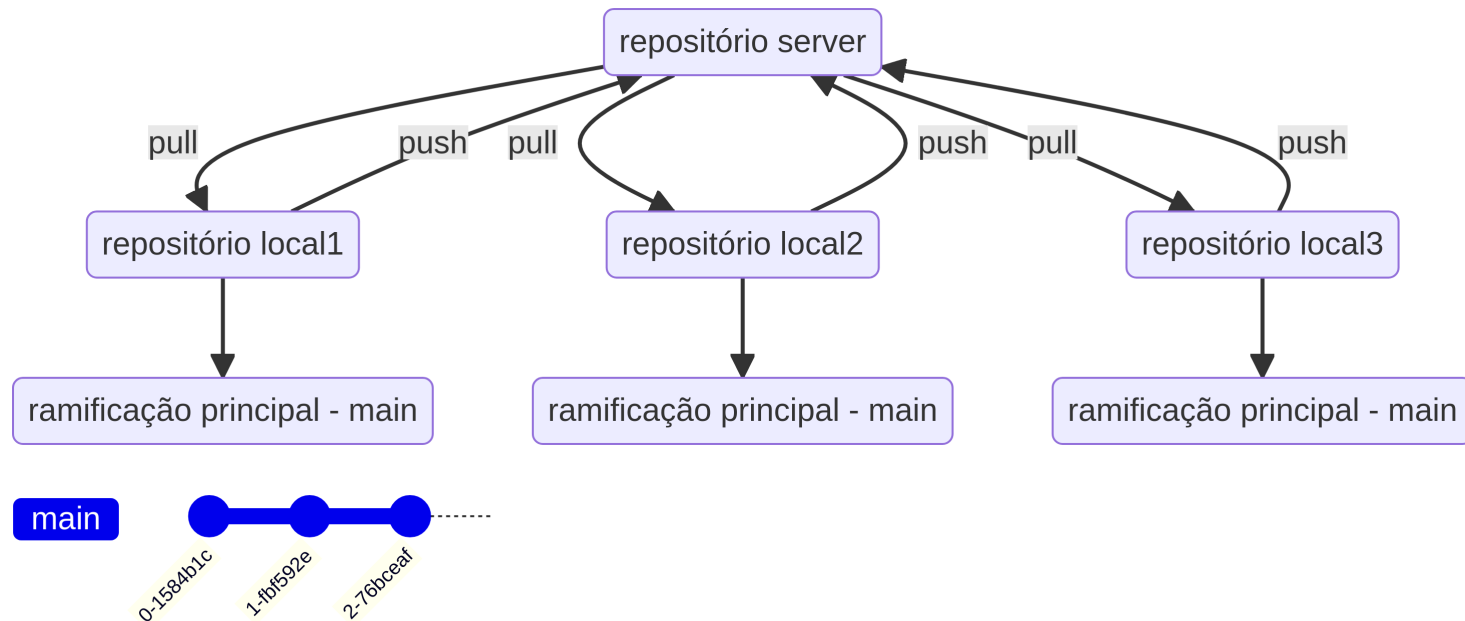
- [nuvem] Quando um repositório é criado com GitHub, ele é armazenado remotamente na .
- [clonar] Você pode clonar um repositório para criar uma cópia local em seu computador e depois usar o Git para sincronizar os dois.

Termos do GitHub

Committing (git commit) e pushing (git push)

- [commit] Confirmar e enviar são os passos necessários para adicionar as alterações feitas em sua máquina local ao repositório remoto no GitHub.
- [push] Depois de ter um commit ou vários commits prontos para adicionar ao seu repositório, você pode usar o comando push para adicionar essas alterações ao seu repositório remoto.

Fluxo de trabalho simples



```
git pull (atualiza repositório local)
-- criacao de funcionalidade
git commit -m "feature: Adicionando um botao para upload de arquivo"

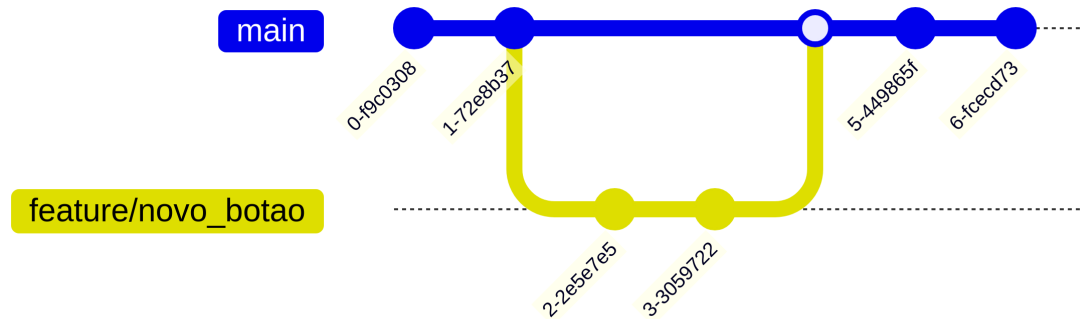
-- corrigindo uma nova funcionalidade
git commit -m "fix: Correção da função upload_files"
git push (atualiza repositório server)
```

Termos do GitHub

Ramificações (git branch)

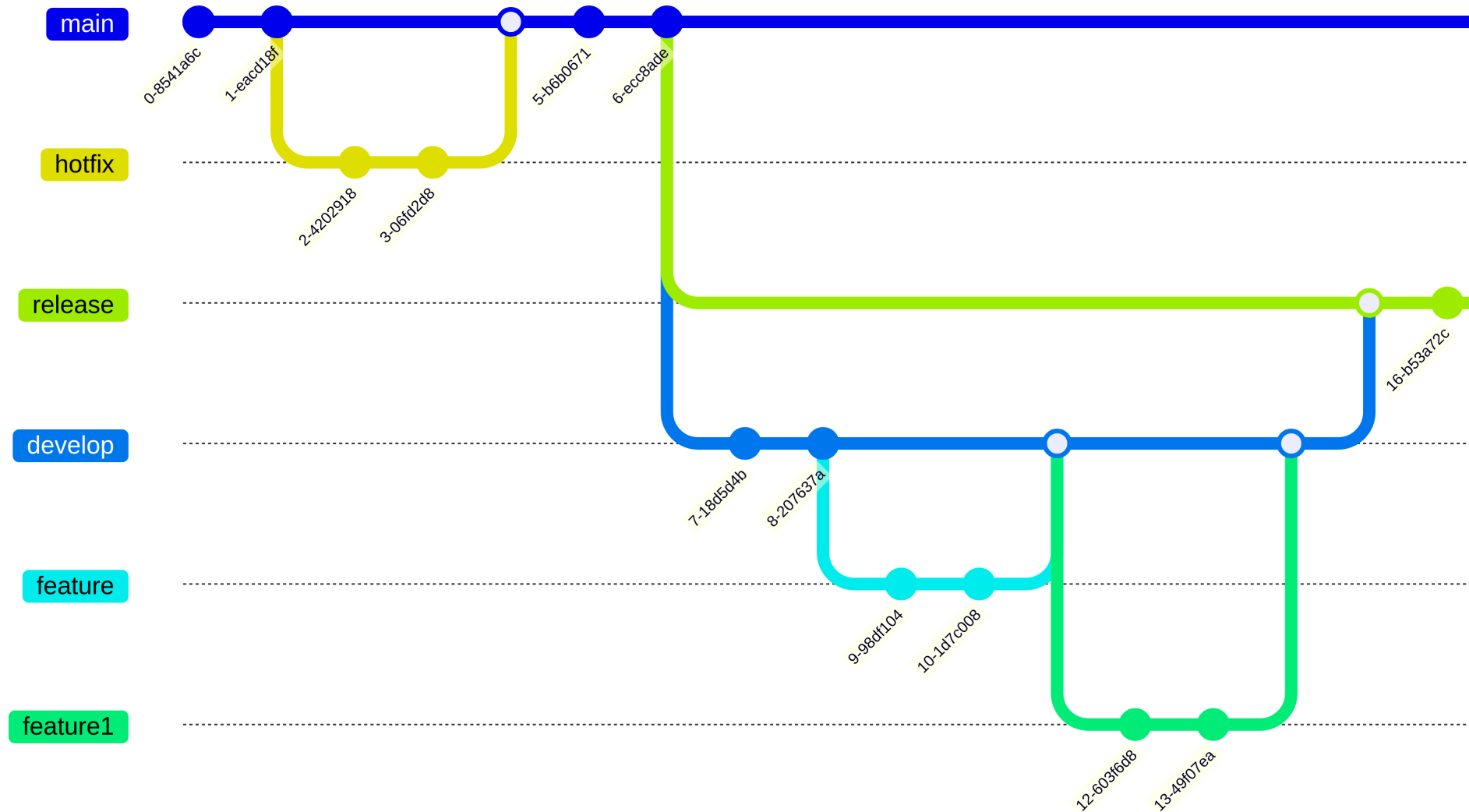
- [branch] Você pode usar ramificações no GitHub para isolar o trabalho que ainda não deseja mesclar em seu projeto final.
- [funcionalidades] As ramificações permitem que você desenvolva recursos, corrija bugs ou experimente novas ideias com segurança em uma área contida do seu repositório.
- Normalmente, você pode criar um novo branch a partir do branch padrão do seu repositório.

Fluxo de trabalho com ramificações simples



```
git commit
git commit
git branch feature/novo_botao
git checkout feature/novo_botao
git commit
git commit
git checkout main
git merge feature/novo_botao
git commit
git commit
```

Fluxo de trabalho com ramificações complexas



Mão na massa

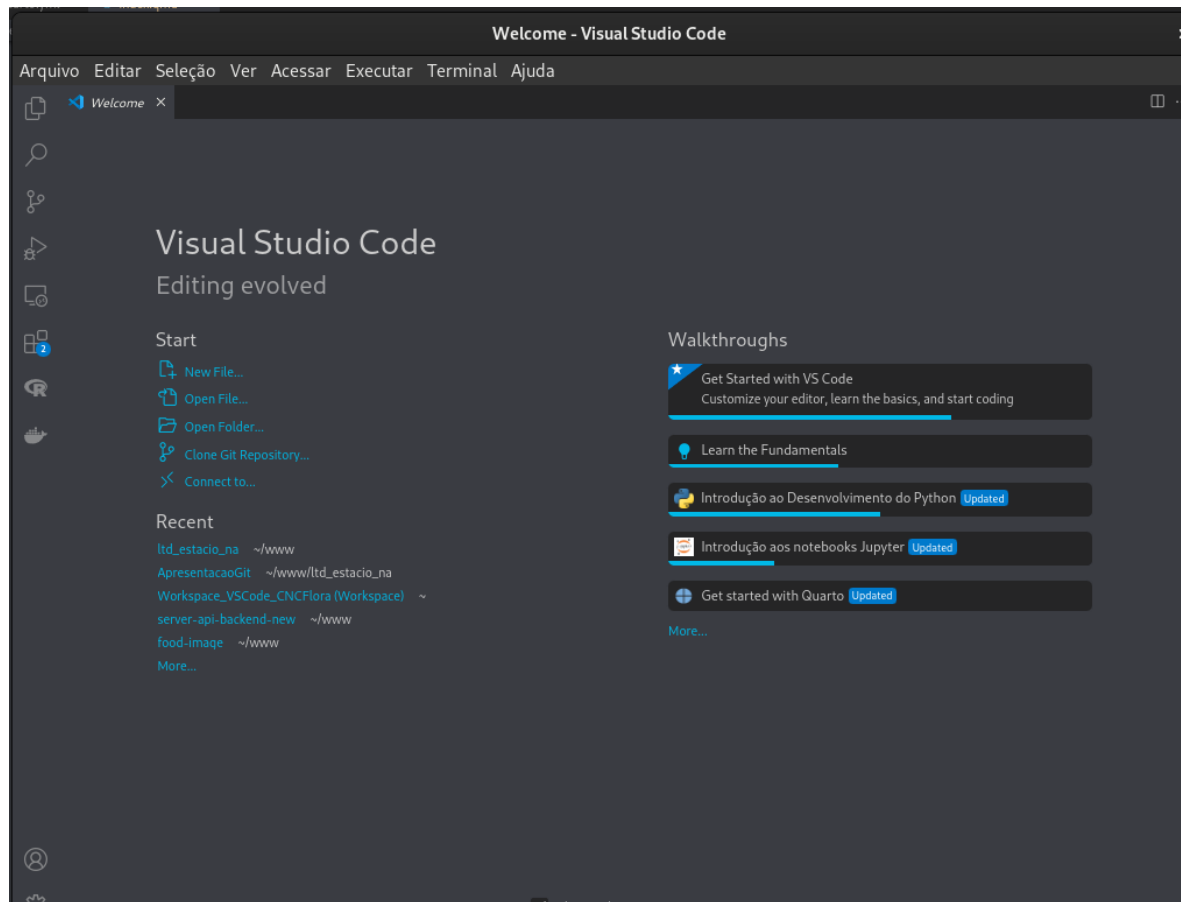
Instalando o vscode:

<https://code.visualstudio.com/download>

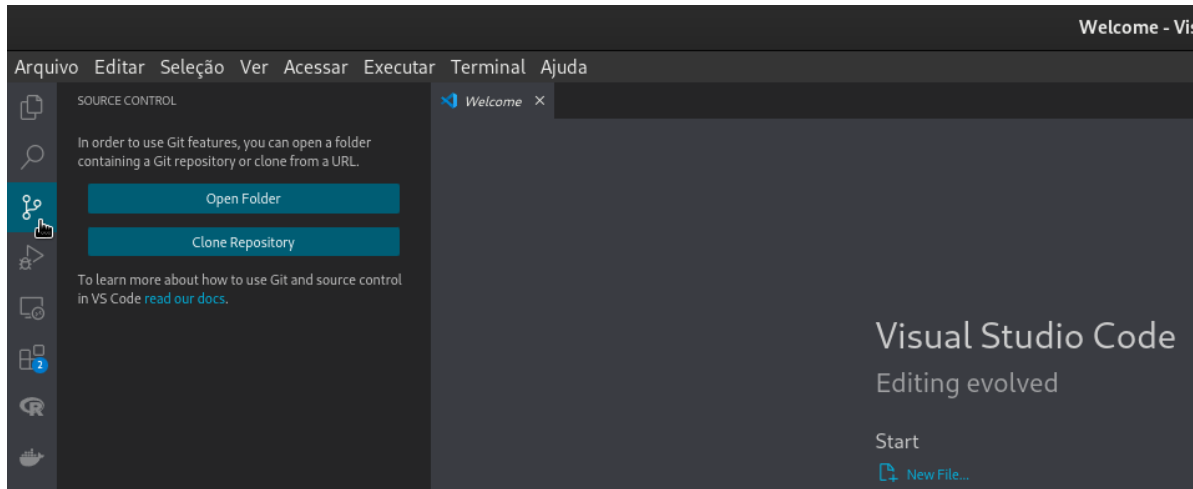
Crie uma conta no github:

<https://classroom.github.com/a/jb8umw-J>

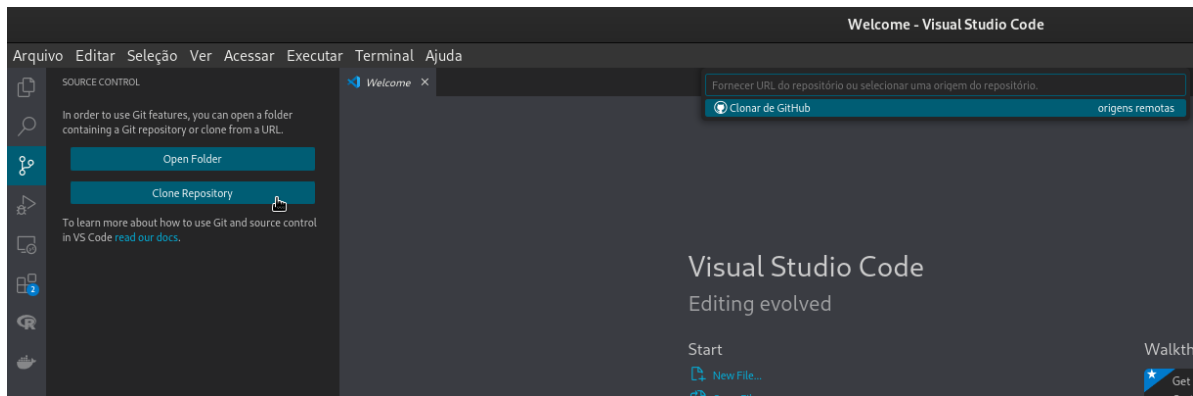
1 - Tela principal do VSCode



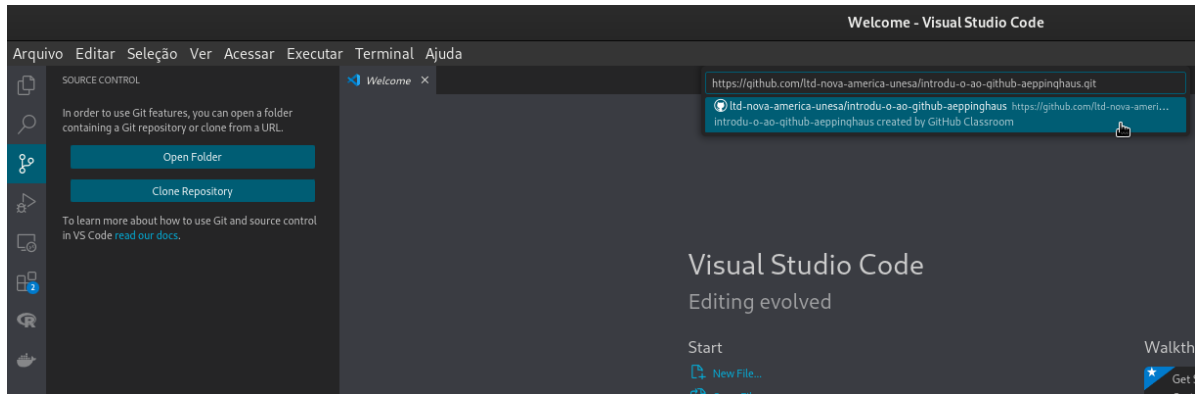
2- Ícone do gerenciador de versão Git



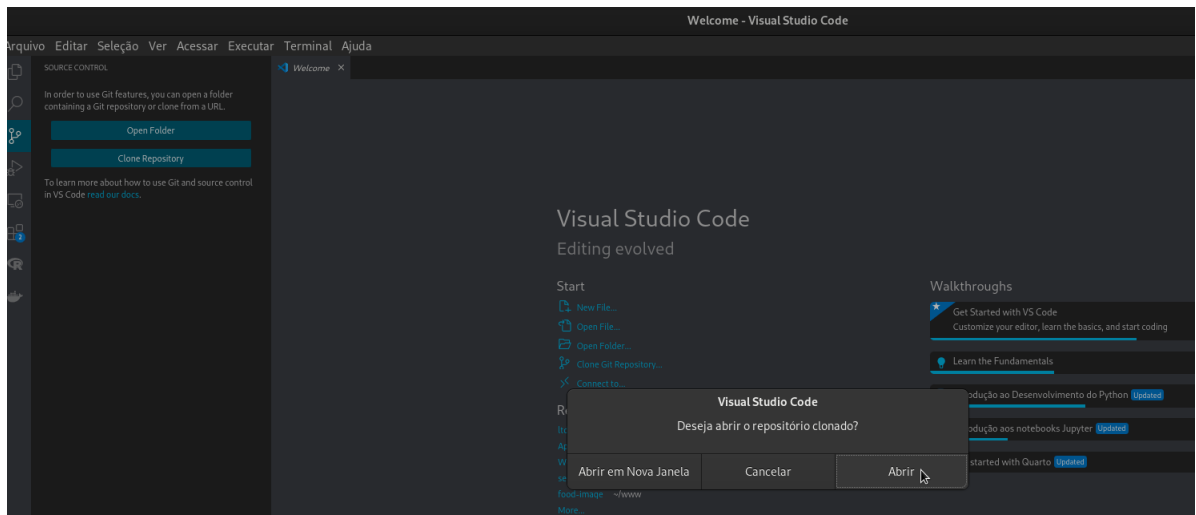
3 - Clonando o repositório do GitHub



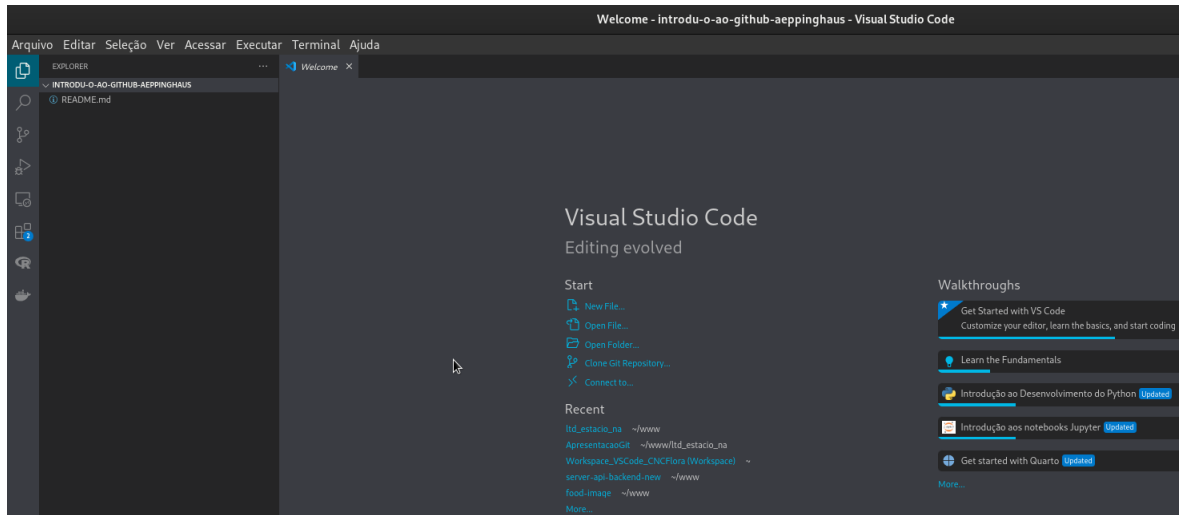
4 - Cole o nome do projeto



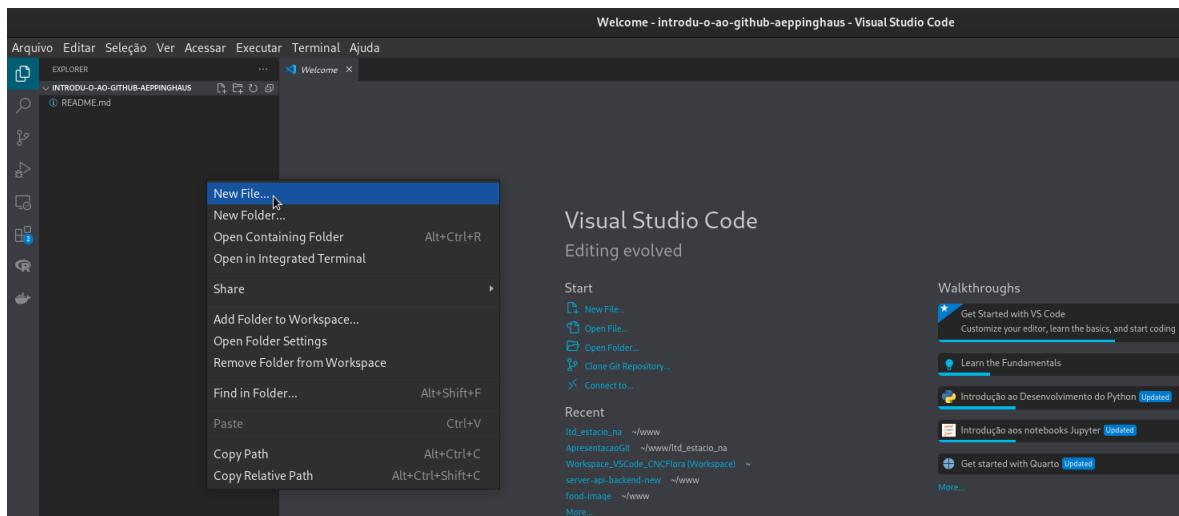
5 - Abra o repositório clonado



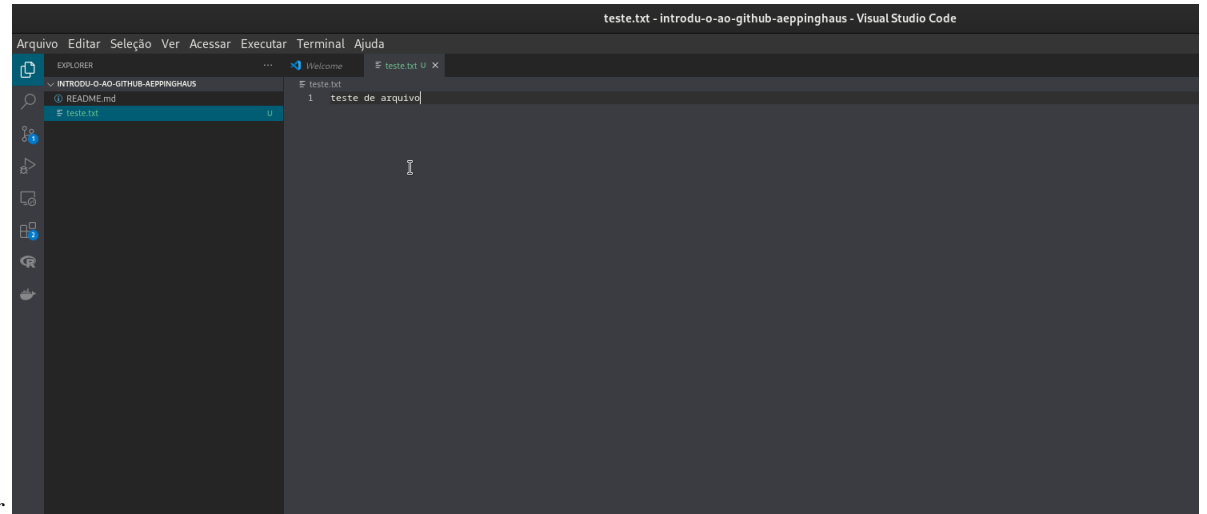
6 - Tela com o projeto importado



7 - Com o botão direito do mouse clique em New File (novo arquivo)

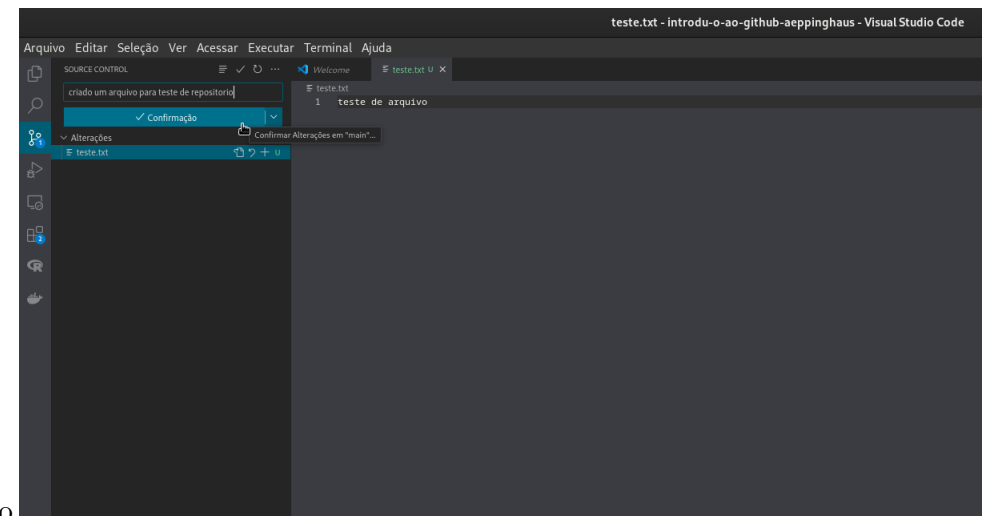


8 - Crie um arquivo com seu nome e extensão (.txt)



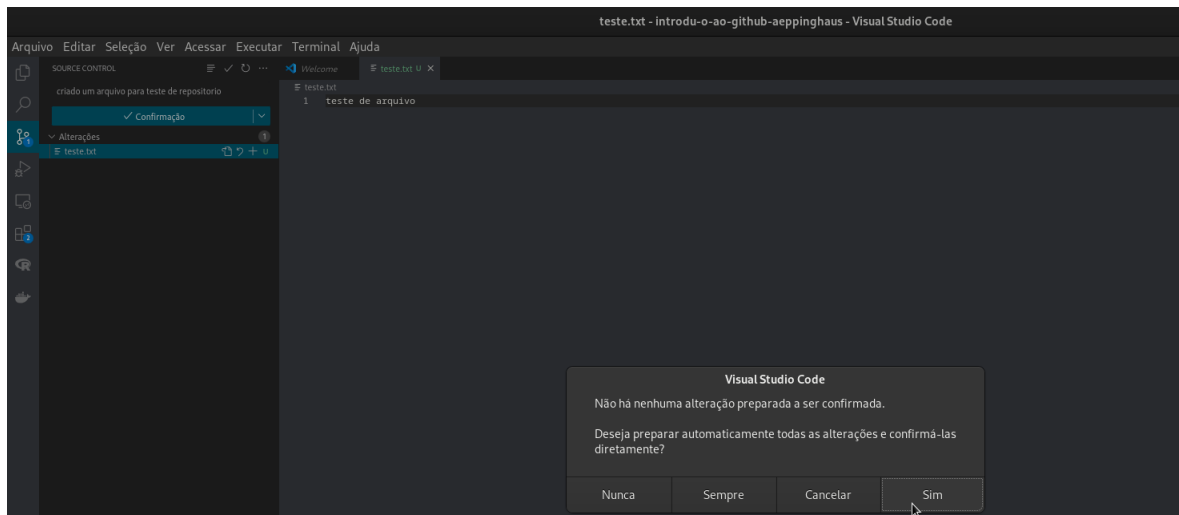
Exemplo: andre.txt e digite um texto qualquer

9 - Clique no icone do git

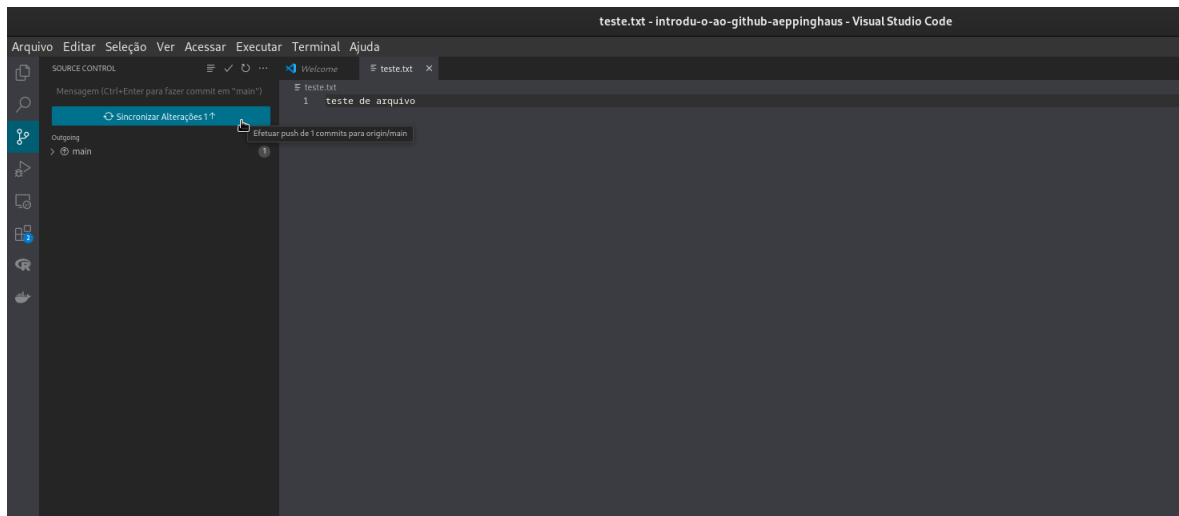


E confirme o texto, adicione uma mensagem para identificar sua alteração

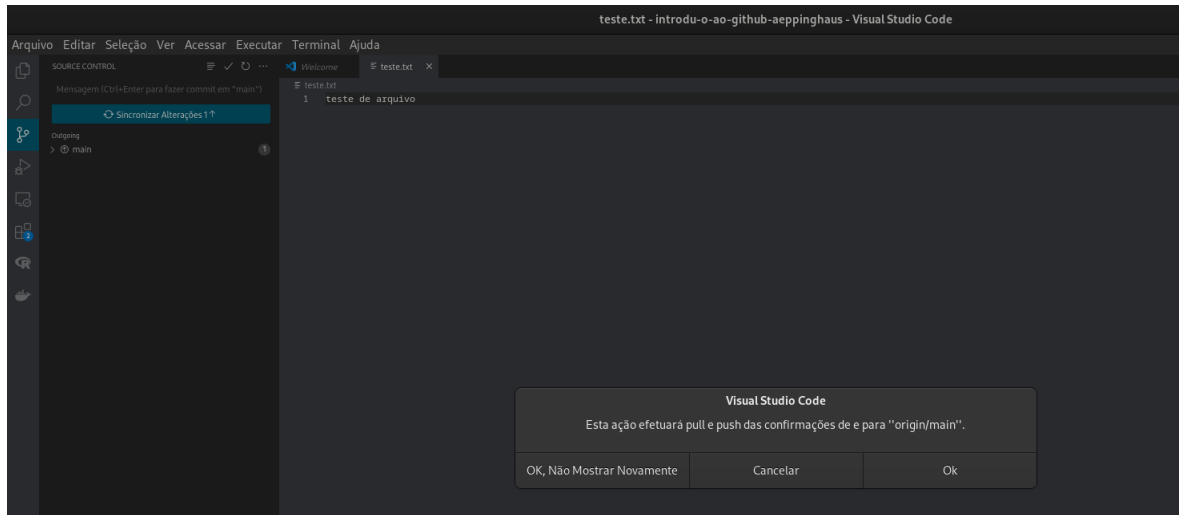
10 - Confirme as alterações



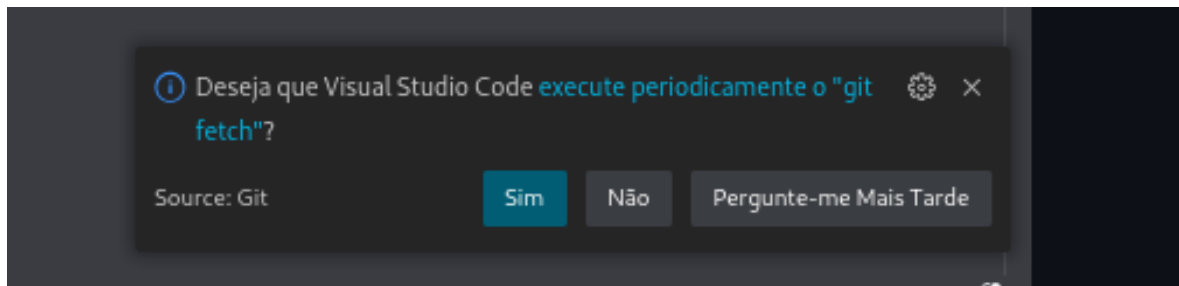
11 - Sincronize com o repositório na nuvem



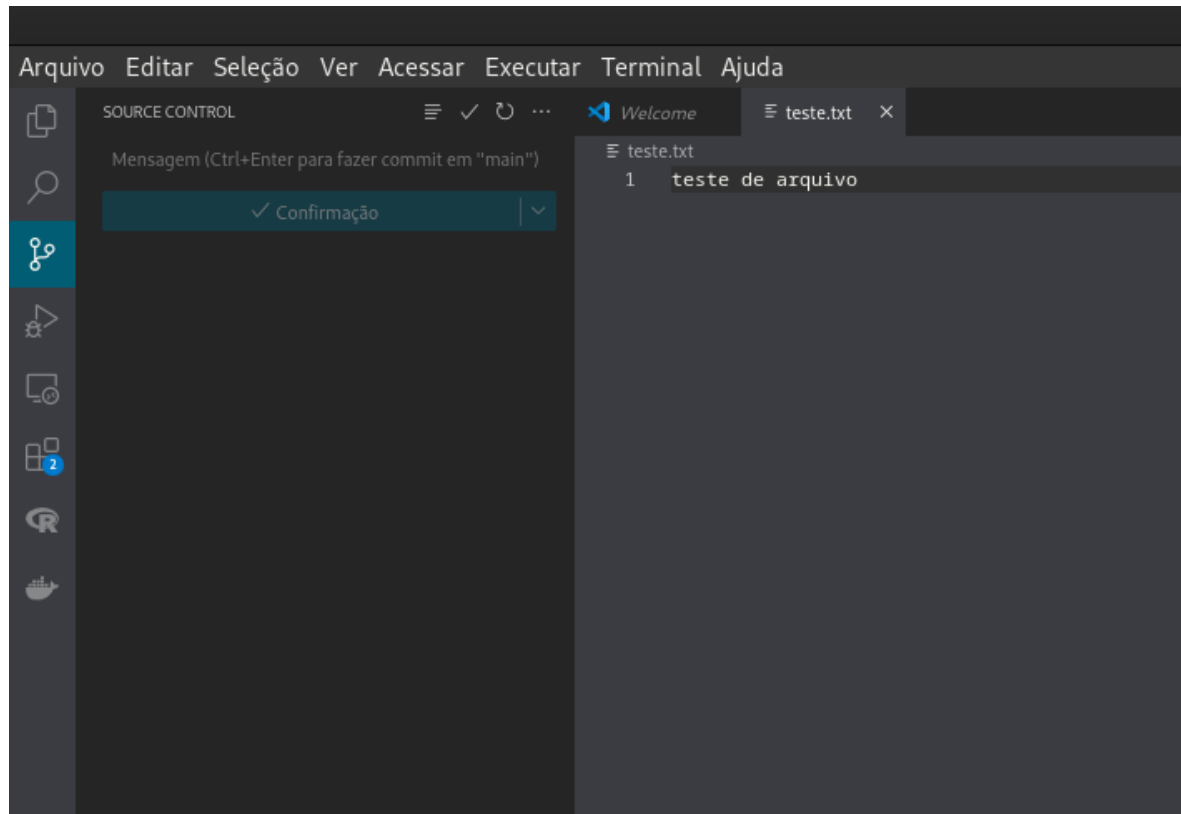
12 - Confirme a sincronização



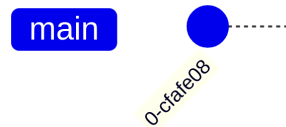
13 - Nesta tela pode clicar em sim



14 - Pronto, o arquivo foi sincronizado na nuvem



15 - Resultado



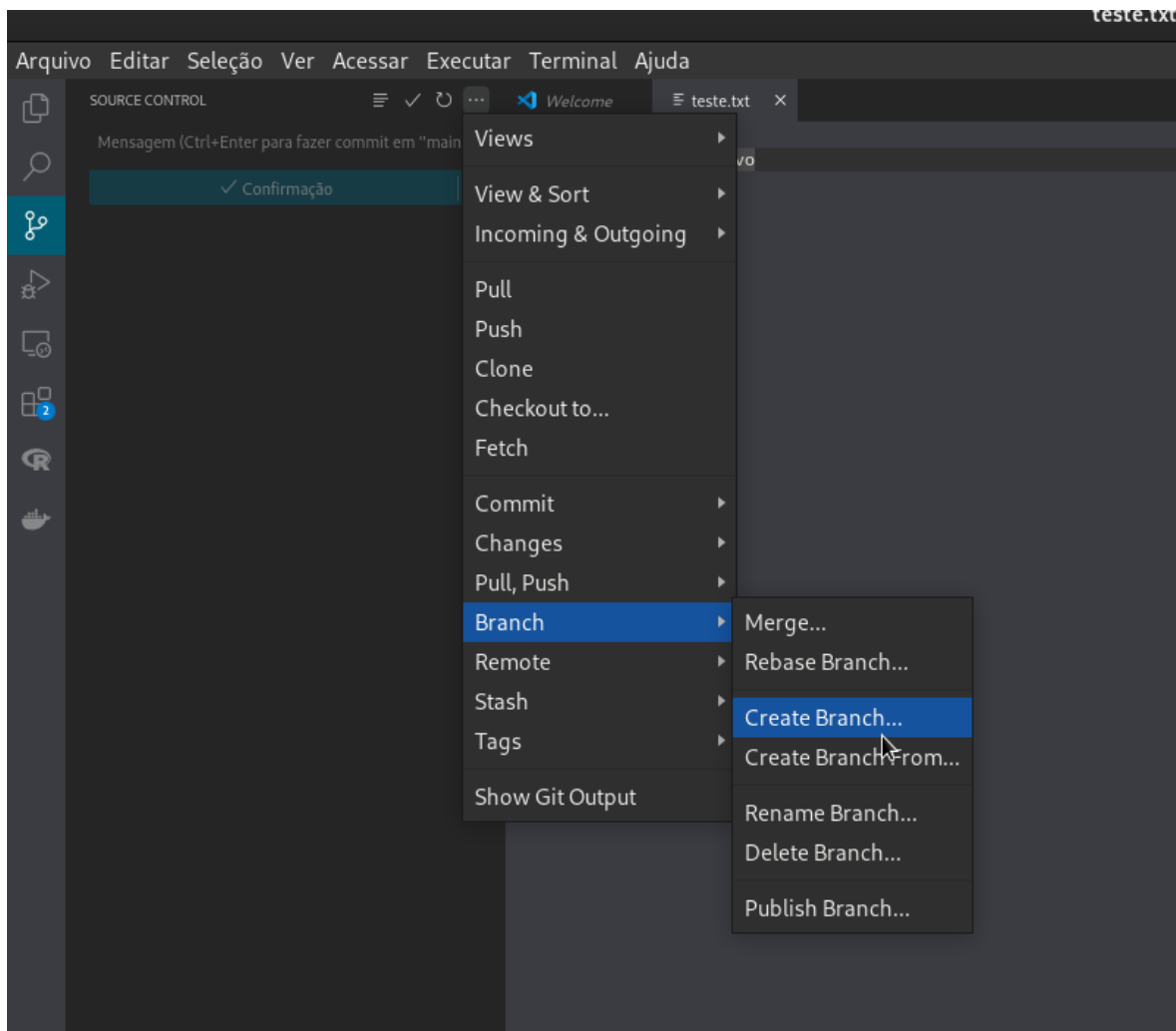
16 - Abra o seu site de projeto no GitHub



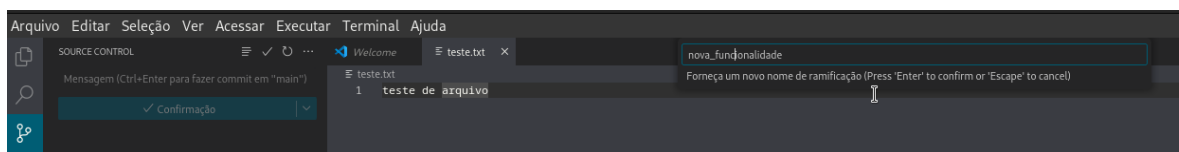
E visualize as alterações

17 - Vamos criar uma ramificação

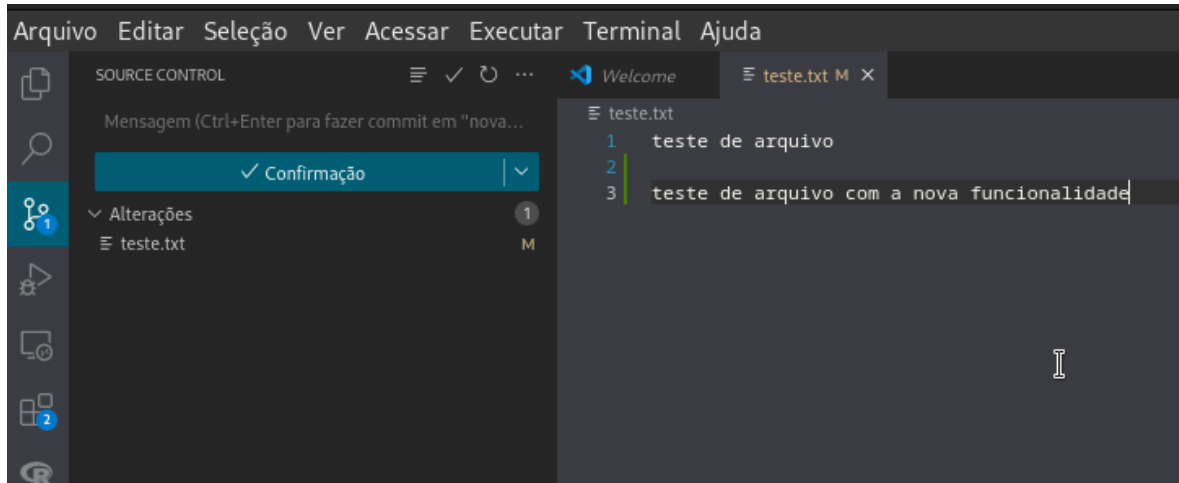
Clique no ícone com três pontos (...) dentro do ícone de controle do Git e escolha Branch/Create Branch



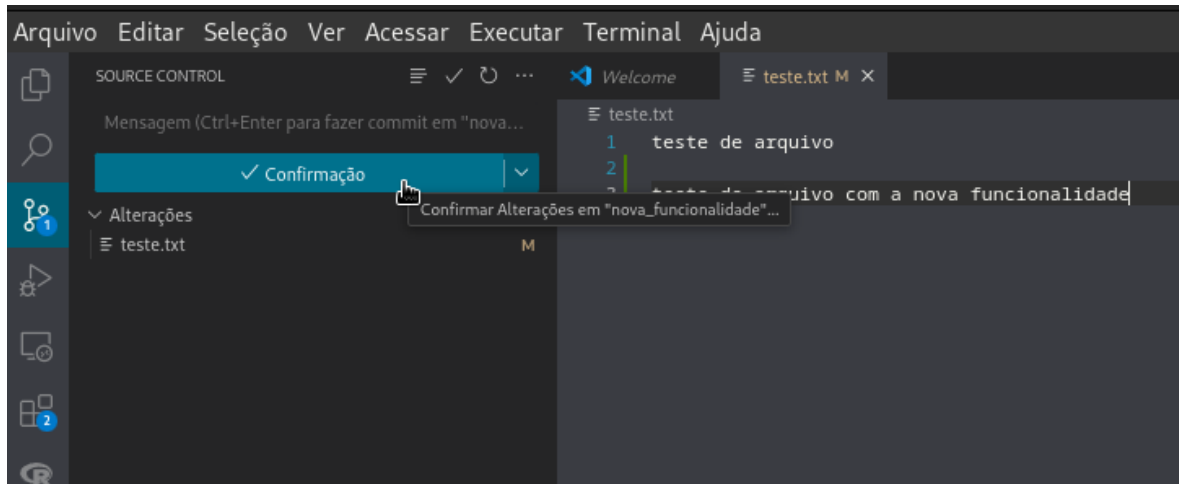
18 - Escolha o nome do branch nova_funcionalidade



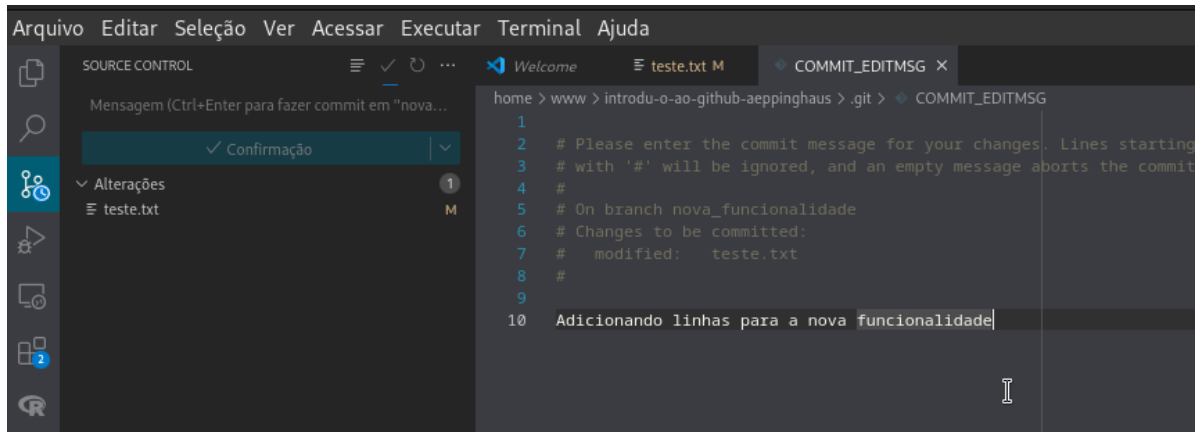
19 - Faça uma alteração no arquivo (.txt)



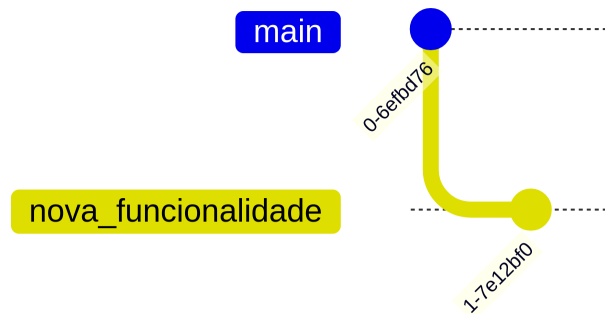
20 - Confirme a alteração



21 - Digite uma mensagem para identificar sua mudança

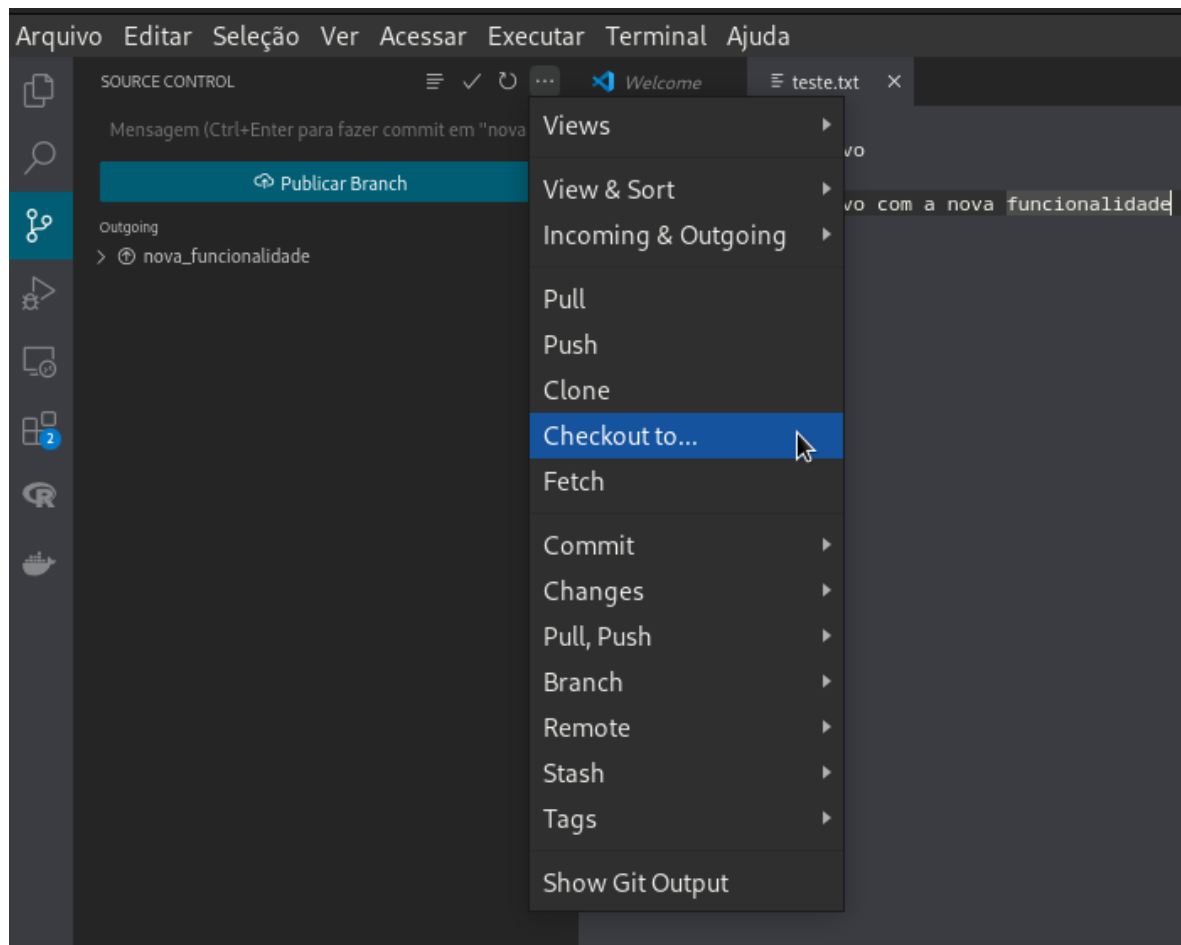


22 - Resultado

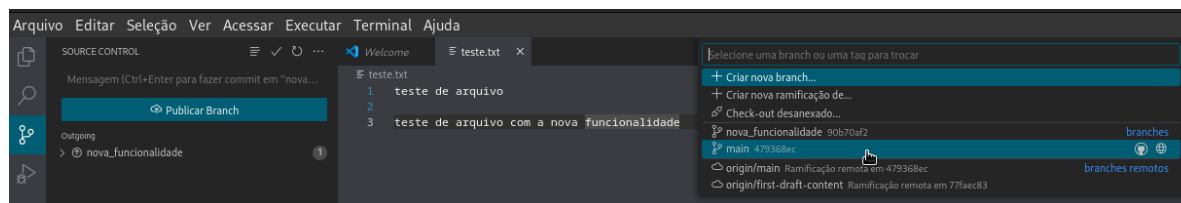


23 - Novamente clique no ícone com três pontos (...)

Escolha Checkout to para sair da ramificação nova_funcionalidade

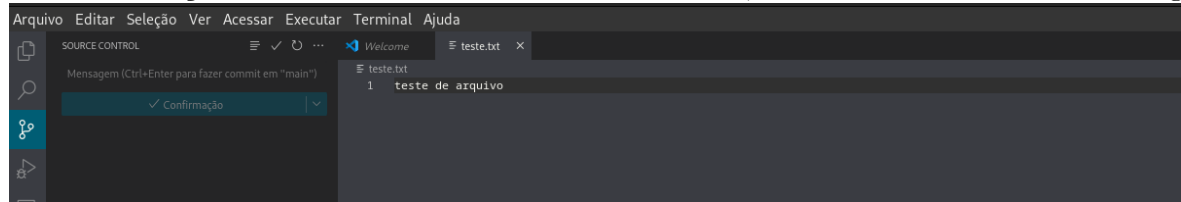


24 - Escolha a ramificação main

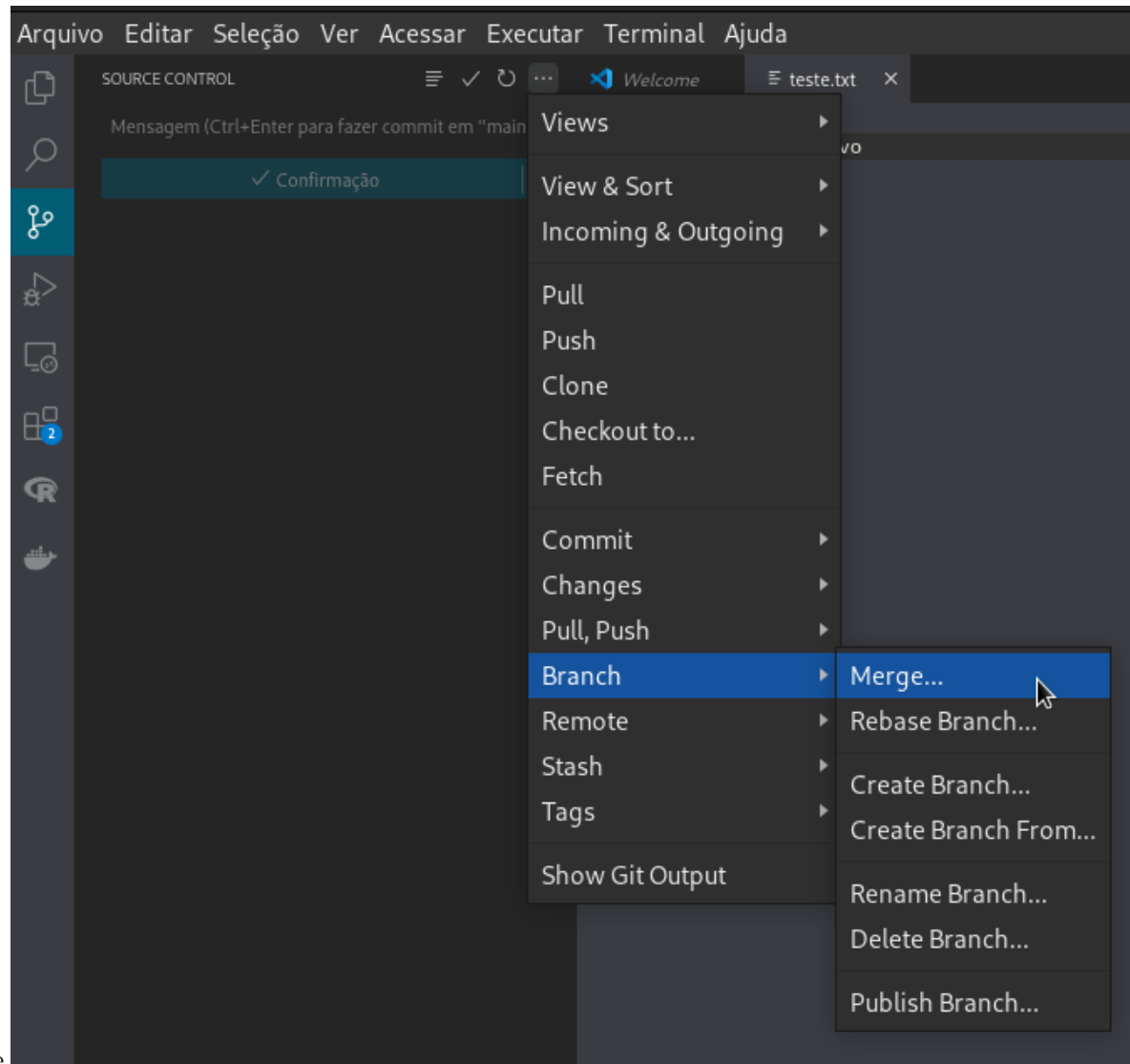


25 - Perceba que sua alteração não existe

Esta ramificação está diferente da nova funcionalidade, será necessário realizar o merge

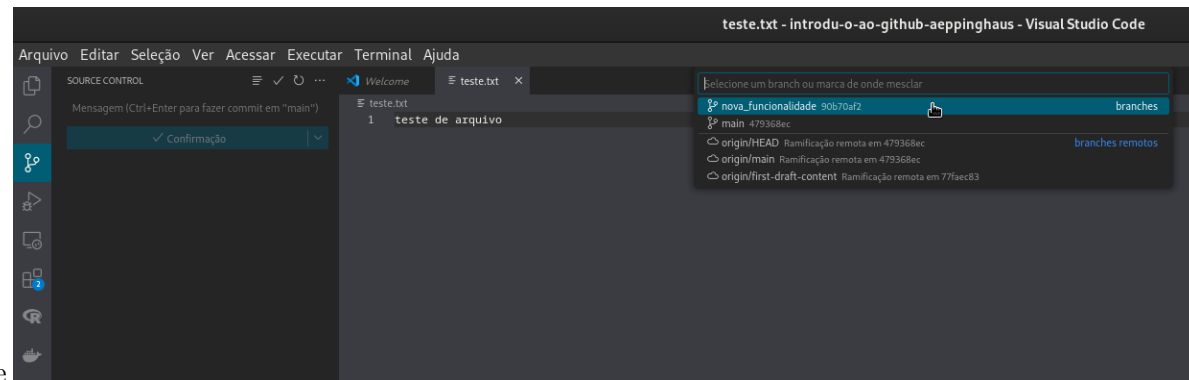


26 - Clique no ícone com três pontos (...)



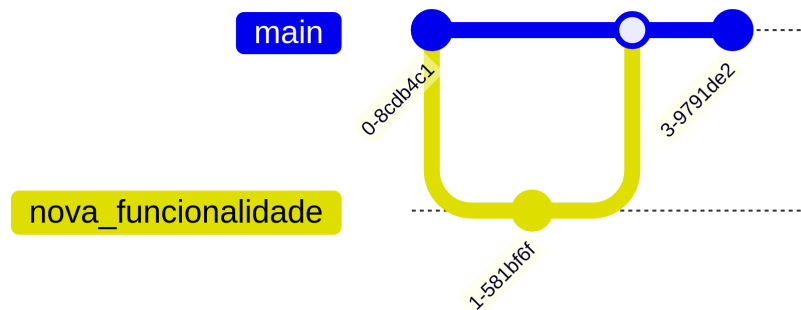
Escolha Branch/Merge

27 - Escolha a ramificação de origem

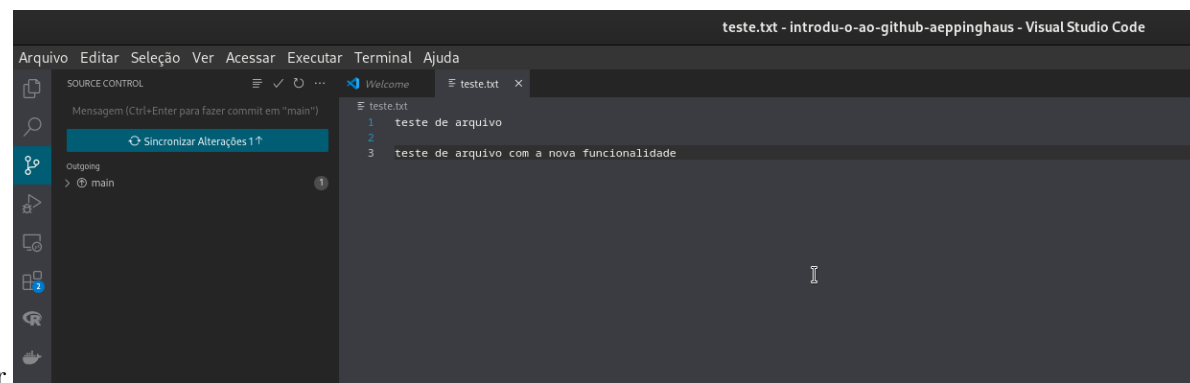


Chamada nova_funcionalidade

28 - Resultado

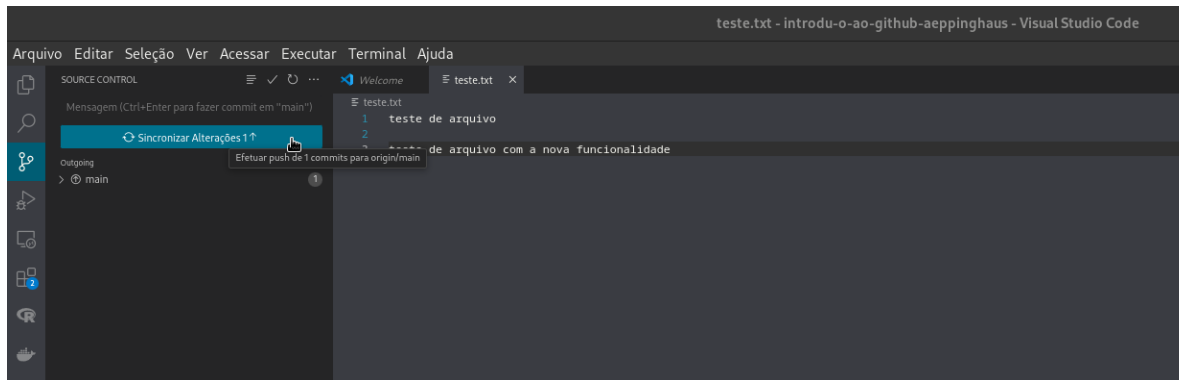


29 - Pronto, as duas ramificações estão idênticas

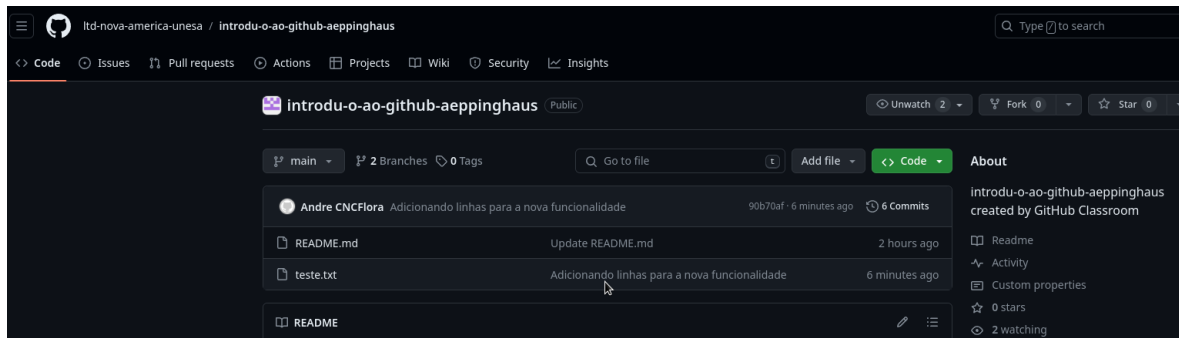


Aparecerá a opção de sincronizar

30 - Clique no botão de sincronizar

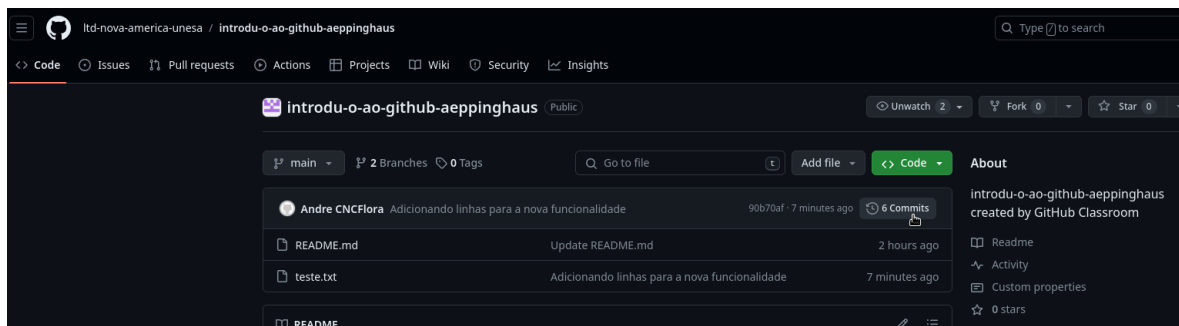


31 - Abra novamente o projeto na web e confirme o envio

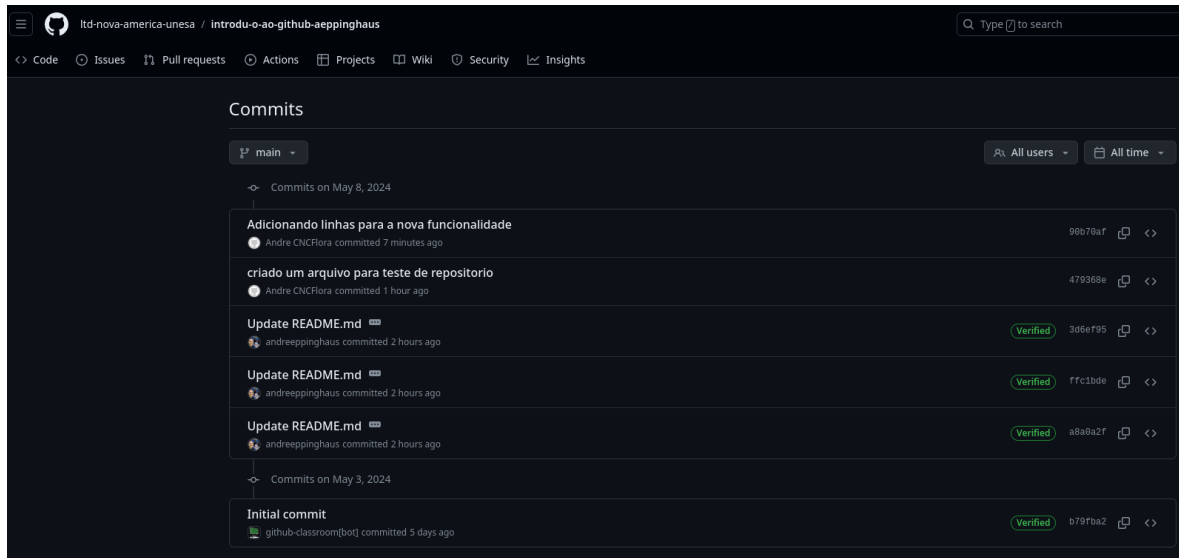


32 - Clique no botão commits

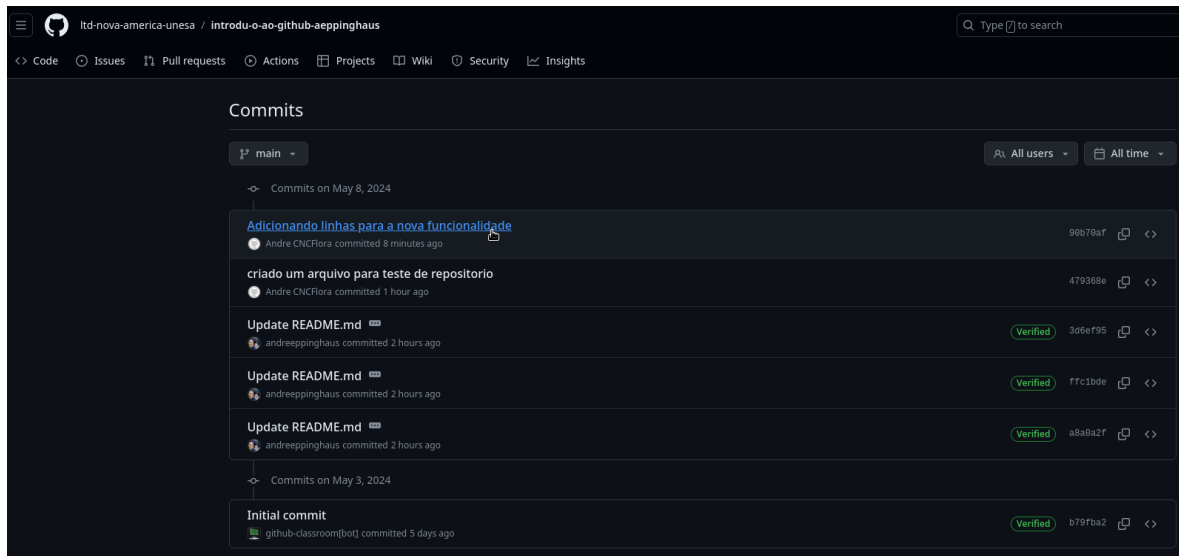
Para visualizar todas as alterações



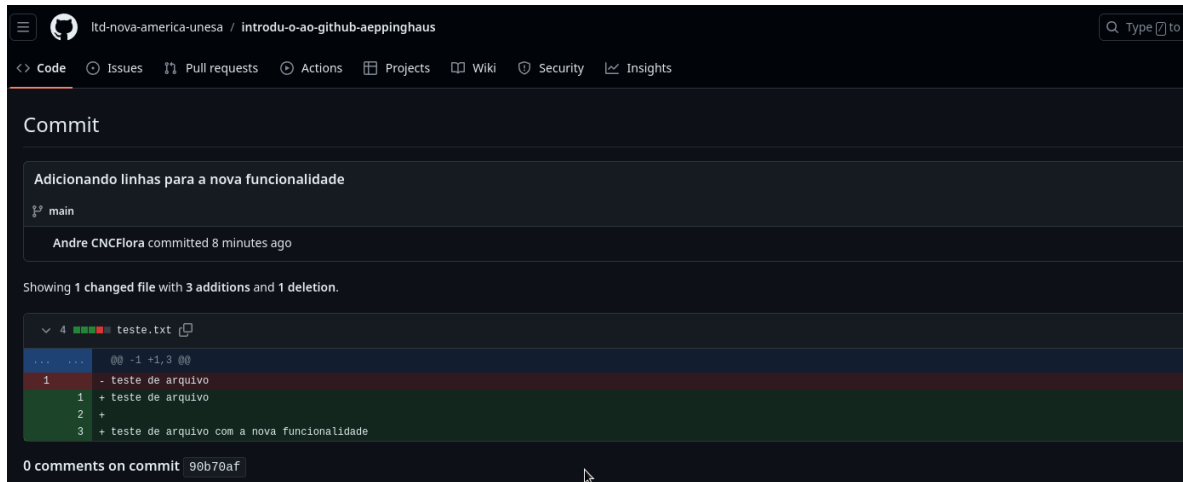
33 - Além da mensagem você pode clicar no link da última alteração



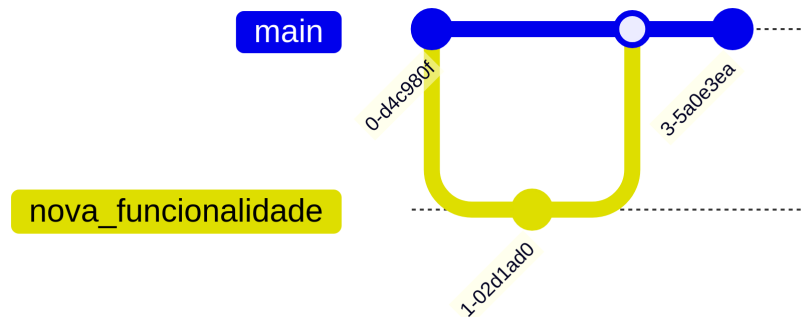
34 - Escolha a última alteração



35 - Irá aparecer todas as alterações realizadas no documento



36 - Resultado final na nuvem



Bibliografia

<https://learn.microsoft.com/pt-br/contribute/content/git-github-fundamentals>

<https://docs.github.com/pt/get-started/using-github/github-flow>